

Fragensammlung CG VO

Alle Angaben in diesem Dokument sind ohne Gewähr auf Richtigkeit oder Vollständigkeit. Gerne kontrollieren, korrigieren und erweitern!

INFO: Zur Beantwortung der Fragen wurden die Vorlesungsaufzeichnungen aus SS2021, die originalen Foliensätze der LVA und die Zusammenfassung "CG Zusammenfassung 2020.pdf" aus dem VOWI verwendet. Der Großteil der Fragen stammt aus früheren Prüfungen, der Rest wurde selbst ausgedacht.

Welche Themen wurden in der Vorlesung behandelt? (wichtigste Punkte)

1. Graphics Programming

- a. Graphical Data Processing
- b. Graphics Programming Levels & APIs
- c. Szenengraph
- d. Unity3D
- e. Shaders

2. Advanced Modeling 1

- a. Motivation: Komplexe Geometrien erzeugen, die nicht durch Meshes repräsentiert werden können
- b. Partikelsysteme
- c. Implizite Modellierung
 - i. Blobby Objects
- d. Prozedurale Modellierung
 - i. sweeps
 - ii. cellular texture generation
 - iii. knitwear
- e. structure deforming transformations

3. Advanced Modeling 2 - alles zu Kurven

- a. parametric curves
 - i. Bezier-curve
 - ii. Bernstein-Polynome
 - iii. de Casteljau-Algorithmus
- b. polynomial curves
 - i. B-Spline
 - ii. Evaluation of B-Spline curves (de Boor)
- c. rational curves
 - i. rational bezier curves
 - ii. NURBS
- d. Tensorproduktflächen
 - i. Bezierflächen
 - ii. B-Splineflächen
 - iii. Bezier Triangles
- e. Subdivision surfaces (Loops scheme)

4. Advanced Data Structures

- a. Motivation: Extrem hohe Anzahl von Daten erfordert neue Datenstrukturen, die Verarbeitung/Suche/andere Operationen ermöglichen

- b. Punktwolke
- c. Wire-frame-model
- d. BSP Tree
- e. kd Tree
- f. Octree
- g. CSG Tree
- h. Bintree
- i. Grid

5. Sampling and Reconstruction

- a. Tools for Sampling and Reconstruction
 - i. Fouriertransformation
 - ii. Faltung
 - iii. Faltungstheorem
 - iv. Filtern
- b. Sampling
- c. Reconstruction
 - i. Samplingtheorem
 - ii. Praxisbeispiele

6. Digital Fabrication

- a. TODO

7. Texturing

- a. Mapping & Parametrization
- b. Aliasing
- c. Anti-Aliasing
 - i. Summed Area Table
 - ii. MIP Mapping
 - iii. Anisotropic filtering
- d. Texturmapping-Techniken
 - i. Bump mapping
 - ii. Horizon mapping
 - iii. Parallax mapping
 - iv. Relief mapping

8. Computational Photography

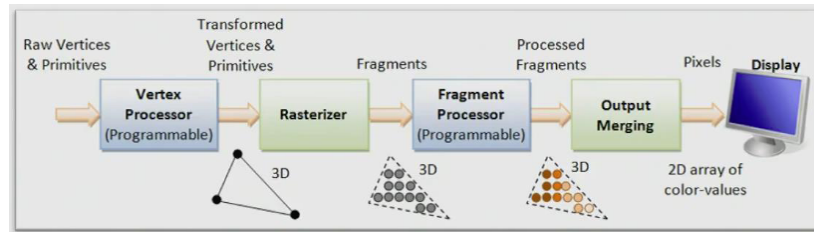
- a. Unterschied zu digital Photography
- b. Image stacks
- c. Gigapixel images
- d. Deep photo
- e. Photography basics
- f. Image filtering
 - i. Bilateral filtering

Rasterisierung

- Was ist das?
 - Diskretisierung, bei dem man aus Dreiecksprimitiven ein Pixelmuster erzeugt. Das Ergebnis der Rasterisierung sind Fragments, die vom Fragment Shader

verändert werden können.

- Wie kommt man von Vertices zu einem Ausgabebild?
 - Indem man die Rendering Pipeline durchläuft



Implicit Modeling

- Wie sehen implizite Funktionen aus?
 - Unbekannte Variablen kommen in der Gleichung auf der gleichen Seite vor, der Threshold auf der anderen Seite
 - Implicit equation e.g., $f(x, y) = -(x^2 + y^2) = T$
 - Vs. explicit equation e.g., $y = kx + d$
- Wie erhält man die Oberfläche eines impliziten Modells?
 - Die Oberfläche stellt alle Punkte dar, die die implizite Funktion erfüllen
 - ist nicht explizit gegeben, sondern wird nach einer Simulation bekannt
- Beispiele für Implicit Modeling?
 - Blobby Objects

B-Splines

- Was sind B-Splines?
 - Streckenweise Polynomiale Kurven, ähnlich wie Bezier-Kurven
- unterschied zu Bezier-Kurven?
 - Bei Bezier-Kurven haben alle Kontrollpunkte Einfluss auf die Kurve (durch die Bernstein-Polynome)
 - Anstatt dass alle Bernstein-Polynome Einfluss haben, haben nur bestimmte Kontrollpunkte Einfluss, abhängig vom Grad der Basisfunktion N.
 - Der Grad der Gewichtsfunktion ist unabhängig vom Grad der Kontrollpunkte
 - Lokaler Einfluss der Kontrollpunkte: Zusätzlicher Parameter k, der den Grad der Gewichtsfunktion bestimmt
- Methode zum Approximieren von B-Splines?
 - Knot insertion: Die Kurve kann hierarchisch approximiert werden, indem man die Linie so lange zerlegt, bis die Konvexe Hülle einer Linie entspricht

Sampling und Rekonstruktion

- Was bedeutet "Sampling"?

- Das Diskretisieren eines kontinuierlichen Signals
- Wie sampelt man richtig?(Nyquist Frequenz):
 - Indem man über der Nyquist-Frequenz abtastet. Die Nyquist-Frequenz ist das doppelte der Bandweite im Frequenzraum
 - Indem man eine Kammfunktion richtig wählt: Multipliziert man das ursprüngliche Signal mit einer Kammfunktion, die zu große Abtastintervalle hat, ist der Abstand der Käme in der fouriertransformierten Kammfunktion zu klein. Die Ursprungsfrequenz überschneidet sich dann mit Schattenfrequenzen -> Exakte Rekonstruktion nicht möglich.
- Wie rekonstruiert man?
 - Man kann ein Grauwertspektrum/Signal nur dann exakt rekonstruieren, wenn die Frequenz im Frequenzraum bandlimitiert ist, d.h. wenn keine Frequenzen außerhalb des Intervalls $[-u, u]$ vorkommen.
 - Man wendet eine Boxfunktion im Frequenzraum an, um alle Schattenspektren zu eliminieren, die beim Sampling durch die Faltung der fourier-transformierten Kammfunktion mit der Ursprungsfrequenz entstehen.
 - Eine exakte Rekonstruktion ist durch die sinc-Funktion möglich. Dazu muss man das diskretisierte (gesampte) Signal im Ortsraum mit der Sinc-Funktion falten -> ist nicht praktikabel, da sinc-Funktion unendlich.
 - Eine approximierende, praktische Rekonstruktion ist mit der Dreiecksfunktion (lineare Interpolation) oder NN-Interpolation (Faltung im Ortsraum mit Boxfunktion).
- Warum wendet man die Fourier Transformation an?
 - Dient dazu vom Ortsraum in den Frequenzraum zu wechseln und mit der inversen Fouriertransformation zurückzuwechseln.
- Welche Filter kommen wo und wann zum Einsatz und warum?
(nicht sicher, ob diese "Filter" gemeint sind)
 - Box-filter:
Bei Faltung im Ortsraum -> NN-Interpolation zur Rekonstruktion
Bei Multiplikation im Frequenzraum -> Originale Spektrum von Schattenspektren isolieren
 - Tent-filter:
Bei Faltung im Ortsraum -> lineare Interpolation (alle diskreten Werte direkt miteinander verbinden)

Parallax Mapping

- Was ist das?
 - Technik um Oberflächenunebenheiten von Texturen einen besseren Tiefeneffekt zu geben.
- Funktionsweise

- Man speichert eine Höheninformation zusätzlich zur Textur. Je nachdem aus welchem Winkel man auf die Textur sieht, wird der Viewray einmal mit dem Oberflächenwert geschnitten und darunter mit dem Höhenwert. Man nimmt den Oberflächenwert zur Darstellung der Textur von diesem Blickwinkel, der sich direkt über dem geschnittenen Höhenwert befindet, hat somit ein Offset, das diesen Parallax-Effekt erzeugt.
- Sieht man von 90 Grad auf die Textur, ist der Offset 0, da der Schnittpunkt von Oberflächenwert und Höhenwert an der gleichen Position sind -> kein Parallax-Effekt.
- was ist der Unterschied zu Normalmapping?
 - Normalmapping speichert keine zusätzliche Höheninformation, sondern nur Normalen (Hochaufgelöste Schattentextur wird auf niedrige Anzahl an Dreiecken angewandt)

B-Splines

- was ist das?
 - Streckenweise polynomielle Kurven
- wie unterscheiden sie sich von Bezier-Kurven?
 - Durch ihren lokalen Einfluss: Bei der Bezier-Kurve haben alle Kontrollpunkte Einfluss auf die Kurve. Eine B-Spline-Kurve besteht streckenweise aus Bezier-Kurven. Der Einfluss von Kontrollpunkten ist abhängig von einem Grad k limitiert, der den Einfluss auf das Intervall $[u_0, u_n+k)$ beschränkt. Außerhalb dieses Intervalls ist der Einfluss 0.
 - Grad des Polynoms ist unabhängig von der Anzahl der Kontrollpunkte
- Wie kann man B-Splines approximierend zeichnen? (de Casteljau-Algorithmus/Bernsteinpolynome)
 - Bezier-Kurzen mittels de Casteljau
 - B-Spline: Knot insertion - K erhöhen durch Einfügen weiterer Knoten. Ist $k =$ die Anzahl der Kontrollpunkte, ist die B-Spline-Kurve gleich einer Bezier-Kurve

Sampling

- wie funktioniert das?
 - Man hat ein kontinuierliches Signal, das man diskretisiert, indem man es mittels Multiplikation mit einer Kammfunktion abtastet
- wie kann man Sampling mathematisch beschreiben?
(gemeint war Multiplikation für Sampling und Faltung für Rekonstruktion in der Spatial Domain und Faltung für Sampling und Multiplikation für Rekonstruktion in der Frequency Domain)

- Im Ortsraum:
 - Multiplikation mit Kammfunktion zum Samplen
 - Faltung mit Sinc-Funktion(bzw. -Filter) zur exakten Rekonstruktion
- Im Frequenzraum:
 - Faltung mit Kammfunktion zum Samplen
 - Multiplikation mit Box-Funktion (bzw. -Filter) zur exakten Rekonstruktion
- Fourier-Transformation:
 - sinc-Filter fouriertransformiert ist Box-filter im Frequenzraum
 - Kammfunktion mit Abstand T fouriertransformiert ist Kammfunktion mit Abstand $1/T$ im Frequenzraum

KD-Tree

- Was ist das?
 - Tree mit Knoten, die k-dimensionale Werte enthalten (z.B. 3D-Tree-> x,y,z)
 - ist eine Erweiterung/Sonderfall des BSP (Binary Space Partitioning) Tree.
 - Hat in jedem Knoten k-dimensionale Werte
- Wie funktioniert es?
 - Der Raum wird an beliebigen Stellen (z.B. beim gewichteten Mittel auf der x-Koordinate) achsenparallel gespalten

Octree

- Was ist das?
 - Spezialform des kD-Trees
 - Teilung findet immer in X,Y und Z - Richtung achsenparallel in der Mitte statt
- Wie performen Octrees?
 - Transformationen sind schwer zu implementieren.
 - Kombinationen sind leicht, aber Octrees müssen aneinander angepasst werden.
- Wie wird gerendert?
 - Der Subspace, der am weitesten Weg vom viewer ist wird zuerst gerendert

Partikelsysteme

- Was sind Partikelsysteme?
 - Sind Systeme, die Objekte modellieren, die sich über die Zeit verändern (Simulationen)
 - Viele Partikel werden erzeugt, die eine Lebensdauer haben und am Ende gelöscht werden
 - Partikel-Parameter ändern sich über die Zeit:
 - Location
 - Speed

- Appearance
- Partikel können von außen beeinflusst werden
 - Interagieren mit anderen Partikeln
 - z.B. Gravitation, Wind o.Ä.
- Wofür werden sie genutzt?
 - I.A. werden Objekte generiert die sich über die Zeit verändern (z.B. für Simulationen)
 - Flowing, Spattering, Expanding etc.
 - Damit werden natürliche Phänomene modelliert
 - Regen, Schnee, Explosionen, Grasbüschel, Wasserfälle ...

Blobby Objects

- Was sind Blobby Objects?
 - Implizit Modellierte Objekte, die sich je nach Bewegung und Distanz zu anderen Objekten der gleichen Art verändern (sowas wie Kohäsion, abhängig von Schwellwert T)
 - Zur Simulation von z.B. Wassertropfen, Molekülen
- Was bedeutet implizit modelliert?
 - Haben keine fixe Form oder Topologie
 - Dass die Form und Oberfläche nicht von Anfang an gegeben sind.
 - Dass es keine Grenzlinien zwischen gleichen Objekten gibt
 - Differenzierbar
- Wie sind diese definiert (Formel)?
 - Man hat Kontrollpunkte, die Einfluss auf das Objekt haben und dieses "Auseinanderziehen".
 - Man hat einen Ausgangspunkt (Parameter der Funktion), eine Einflussfunktion (Glockenkurve) e^{-r^2} , einen Dichtewert (Abstand zum Ausgangspunkt) r_k .
 - Der Dichtewert r_k wird in die Einflussfunktion (x-Achse) eingesetzt -> Ergebnis des Wertes auf der y-Achse ist der Einfluss des Kontrollpunktes (also wie stark vom Ausgangspunkt weggezogen wird)
 - Die Werte a_k und b_k sind Blobbiness-Faktoren, beeinflussen also die Form (Blobbiness) des Objekts
 - Alle Dichtewerte werden aufsummiert: Das Ergebnis ist ein Dichtewert. Ist dieser Dichtewert kleiner als T, befindet man sich innerhalb des Blobby objects, ist der Wert größer, dann außerhalb.
 - Alle Punkte, bei denen die Dichtewerte den Wert T haben, bilden die Oberfläche des Objekts.
 - Reminder: Bei der impliziten Modellieren bilden genau jene Punkte das Objekt, bei denen die implizite Gleichung gleich T ist.
- Zu welcher Art der Modellierung gehören sie?

- Implizite Modellierung
- Zusammenhang zwischen Blobby Objects und Partikelsystemen erklären.
 - Können mit Partikelsystem kombiniert werden. Ab einer best. Schwellwert-Entfernung können die Objects gelöscht werden.

Superquadrics (3d), Superellipse (2d), Superellipsoid (3d)

- Was ist das?
 - Geometrische Formen
 - Superquadrics: Verallgemeinerte Darstellung von Quadrics mit zusätzlichen Parametern
 - Analog für Superellipse, Superellipsoid

Winged Edge Datenstruktur

- Was ist das?
 - Eine Advanced Data Structure, bei der die Kante das Zentrale Element ist
 - Zu jeder Kante wird ein Start- und Endpunkt gespeichert
 - Wings sind Pointer -> Es werden Pointer gespeichert zu Faces/Ebenen auf beiden Seiten der Kante und zu den benachbarten (vorangegangenen und nachfolgenden) Kanten der Kante
- Unterschied zwischen Winged Edge und B-Reps?
 - **TODO**
- Vorteile der Winged Edge Datenstruktur gegenüber B-Reps.
 - **TODO**

Faltungstheorem

- Was sagt das Faltungstheorem aus?
Wenn man im Ortsraum zwei Signale hat und diese faltet, hat man im Frequenzraum eine Multiplikation der zugehörigen Spektren.
- Was ist eine Faltung?
Anschaulich bedeutet die Faltung $f * g$, dass jeder Wert von f durch das mit g gewichtete Mittel der ihn umgebenden Werte ersetzt wird.
- An welchen Stellen im Ortsraum/Frequenzraum kommt die Faltung vor?
- Welcher Filter wird bei Rekonstruktion optimalerweise verwendet und warum?

Digital Fabrication

- Pipeline erklären.
- Welche Datenstruktur wird verwendet?
- Was versteht man unter Slicing und Supportstrukturen?
- Was ist Stereolithography?

MIP-Mapping

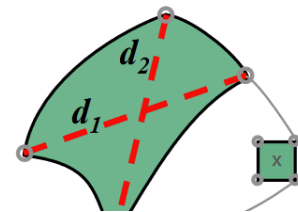
- Warum wird bei MIP Mapping der Logarithmus dualis verwendet und kein anderer?
 - Weil die Textur um den Faktor 2 verkleinert wird
- Wie berechnet man den Index des nötigen Levels?

■ MIP Mapping Algorithm

■ $D := \text{ld}(\max(d_1, d_2))$

■ $T_0 := \text{value from texture } D_0 = \text{trunc}(D)$

"Mip Map level"



Grafikprogrammierung APIs erklären

TODO

Shader erklären

TODO

De Casteljau

- Was ist das? Wozu braucht man das? Wie funktioniert der Algorithmus?
TODO
- Wo in dem errechneten Casteljau Dreieck sieht man die Tangente des gefundenen Punktes?
TODO
- Geht die Kurve bei der Bezier Kurve durch irgendwelche Kontrollpunkte?
TODO

Sampling

- Ortsraum und Frequenzraum Achsen erklären
 - Ortsraum:
 - x-Achse: Scanline
 - y-Achse: Helligkeit/Intensität
 - Frequenzraum:
 - x-Achse: Frequenz
 - y-Achse: Amplitude (und Phaseninformation = Verschiebung)
- Wozu dient die Fourier-Transformation?
 - Wandelt das Eingangssignal in den Frequenzraum um: Modelliert das Eingangssignal als Summe von Sinusschwingungen
- Kammfunktion und fouriertransformierte Kammfunktion erklären
 - Kammfunktion
- Was passiert bei der Faltung von Frequenzspektrum mit Kammfunktion?
 - Die Fourier-Transformation wandelt das Eingangssignal
 - Faltung des Frequenzspektrums mit Kammfunktion führt zur Replizierung des Spektrums in Originalspektrum und Schattenspektren
- Was passiert wenn die Schattenspektren überlappen?

Summed Area Tabelle

- Was ist das?
 - Eine Anti-Aliasing Methode
 - Man nimmt von jedem Texel den linken unteren Abschnitt, summiert alle Werte von dem Abschnitt auf und berechnet sich dann die nötigen Informationen
- Wie viele Schritte braucht man um etwas auszurechnen?
 - 4 Zugriffe auf die Summed Area Tabelle
- Was sind die Annahmen bezüglich des zu texturierenden Pixels?
 - achsenparalleles Rechteck - nicht unbedingt ein Quadrat erforderlich, so wie beim MIP Mapping der Fall

de Casteljau Algorithmus

- Wofür wird er benutzt?
 - Zum Berechnen der Bezier Kurven
- Wie funktioniert er?
 - Man fängt mit einem Kontrollpolynom an und teilt alle benachbarten Punkte

- Wenn man 4 Knoten und 3 Kanten hat, bekommt man durch lineare Interpolation mit faktor T pro Kante einen neuen Knoten
- Was sind die Eigenschaften einer Bézier Kurve?
 - Man hat Kontrollpunkte und einen Parameter t, der von 0 bis 1 geht, mit dem kann man die Kurve bestimmen.
- Wie kann man eine Bézier Kurve an einem ermittelten Punkt in 2 Kurven teilen?
 - Spannendes Dreieck des de Casteljau Algorithmus enthält neue Kontrollpunkte

Anti Aliasing

- Was ist Aliasing?
 - Aliasing sind Fehler bei der Anzeige, die durch schlechtes oder fehlerhaftes Sampling entstehen
- Welche Methoden zu Anti Aliasing gibt es?
 - Summed Area Methode
 - MIP Mapping
 - Anisotropic Filtering
 - RIP Mapping
 - Footprint Assembly
- Was sind MIP Maps?
 - Damit man weit entfernte Texturen ohne Aliasing darstellen kann, benutzt man MIP Mapping. Eine MIP Map ist eine Folge von Rasterbildern mit dem gleichen Motiv mit abnehmender Auflösung. Das kleinste Bild hat immer 1 Texel
- Wie viel Speicherplatz brauchen sie und warum?
 - 1/3 mehr als das Original, berechnet durch geometrische Reihe
- Wie bestimmt man anhand eines Pixels welche MIP Map geladen wird?
 - **TODO**
- Warum wird zur Berechnung der Logarithmus Dualis verwendet?
 - weil man um den Faktor 2 verkleinert

Advanced Data Structures

- Welche gibt es?
 - Point Cloud
 - Wire-Frame Model
 - Boundary Representation

- Binary Space Partitioning Tree
- kD Tree
- Octree
- Was sind BSP Trees?
 - Raum wird aufgeteilt und man schaut wie man auf gewisse Teile in der Szene zugreifen kann
- Wie werden BSP Trees aufgebaut?
 - Es gibt eine Base Plane, die den Raum teilt. Alle Polygone die vor der Base Plane liegen kommen in den linken Unterbaum und alle die dahinter liegen in den rechten Unterbaum
- Was unterscheidet einen kD Tree von einem BSP Tree?
 - Beim kD Tree braucht man kein Polynom, um den Raum zu teilen, sondern man kann sich aussuchen, wo man den Raum teilt
- Wozu braucht man BSP Trees?
 - Rendering Reihenfolge feststellen
- Wie werden BSP Trees abgearbeitet?
 - von Blickpunkt abhängig, erst entgegengesetzte Seite, dann Wurzel, dann Blickpunkt-Seite

Computational Photography

- Was sind Gigapixel Images?
 - Bilder mit einer sehr hohen Auflösung
- Wie kann man sie erzeugen?
 - Sehr viele, kleine, normale Bilder
- Wie kann man sie darstellen?
 - Zusammenfügen von Bildschirmen
 - Mehrere Projektoren hinter einer Leinwand
- Welche weiteren Verfahren für Bilder gibt es?
 - Deep Images
 - Flash/ No Flash Photography
 - Image Stacks
- Was sind Image Stacks und was kann man damit machen?
 - Image Stacks sind Bilder die wenige Momente hintereinander aufgenommen wurden
 - Sie werden verwendet, um z.B. sich bewegende Dinge aus einem Bild zu entfernen (Gehende Menschen auf einem Platz), oder um alle Personen in einem Gruppenfoto in die Kamera schauen zu lassen

Structure-deforming transformations

- Welche Transformationen es gibt?
 - Bend
 - Twist
- Worin unterscheiden sie sich von affinen Transformationen?
TODO
- Welche Nachteile und welchen Vorteil haben nicht affine/lineare Transformationen?
TODO

NURBS

- Was sind NURBS?
 - NURBS = Non-Uniform Rational Basis Spline
Mit denen kann man alle möglichen Formen von einfachen 2D Linien, Kreisen, Bögen bis hin zu hoch komplexen organischen 3D-Freifformflächen und -Volumenkörpern darstellen
- Wie unterscheiden sie sich von normalen B-Splines?
 - Man verwendet zusätzlich noch die Z-Achse (3D)
- Was bedeuten "Non-Uniform" und "Rational"?
 - Non-Uniform: Die Knoten der Kurve dürfen uneinheitlich sein
 - Rational: Statt Polynomen werden **rationale** Funktionen zusammen gereiht
- Welche Eigenschaften hat ein B-spline?
 - B-Splines sind stückweise Polynome und werden mit Knoten unterteilt
 - Sie sind nicht non-zero in dem ganzen Intervall
 - lokal

Sampling und Sampling-Theorem

- Was ist das?
 - Sampling ist das Abtasten eines Signals, um von einem kontinuierlichen Signal zu diskreten Werten zu kommen
 - Man verwendet hier die Comb-Funktion
- Wie findet man die passende Samplingrate?

- Man muss in regelmäßigen Abständen sampeln. Hat man zu wenige Samples, kann Aliasing (Fehler) entstehen, hat man zu viele kann es zu Speicherproblemen kommen.
- Was kann man vom Frequenzraum ablesen?
 - Die Reziprokwerte der Wellenlängen
- Wofür stehen die Achsen im Frequenzraum und im Ortsraum?
 - Amplitude (?)

Computational Photography

- Was ist das?
 - Digitale Bilderfassungs- und Verarbeitungstechniken, die statt optischen Prozessen digitale Berechnungen verwenden. Diese können die Fähigkeiten von Kameras verstärken bzw. Features einbringen, die vorher nicht möglich waren mit normaler Fotografie
- Welche Anwendungsmöglichkeiten der Comp. Photography gibt es?
 - HDR (High Dynamic Range): Das Smartphone macht gleichzeitig mehrere Bilder und passt die Helligkeitswerte für das resultierende Bild an
 - Entfernen von vorbeigehenden Menschen in einer Szene
 - Gruppenfoto: alle schauen korrekt in die Kamera (Bilder werden zusammengefügt)
 -
- Was ist Bilateral Filtering?
 - Kanten nicht verwischen aber Rauschen weichzeichnen

Unity 3D

- Was ist das?
 - Entwicklungsumgebung für Spiele (+ Engine)
 - WYSIWYG Editor
 - C# Scripting
- Was ist der Szenengraph?
 - Hierarchische Struktur, welche die Struktur der Szene beschreibt
 - Zweige aufgeteilt in content und viewing, wo bei content Transformationen leicht vorgenommen werden können und viewing für die Anzeige zuständig ist
- Welche Levels von Graphics Programming gibts?
 - Low-Level (OpenGL, DirectX, Vulkan, Metal)
 - Direkter Aufruf von API-Funktionen

- Entwickelt mit C++, C#, Java, Python..
- Mid-Level (Java3D, Qt3D, bgfx)
 - Abstraktionsschicht über Grafik-API
- High-Level (Unity, Unreal Engine, ParaView)
 - Render und Game Engines
 - Visualisierung-Toolkits

Cellular Texture Generation

- Was ist das?
 - Ein zelluläres Partikel System, das die Geometrie einer Fläche verändert
- Welche Informationen können Zellen beinhalten (was ist ein Zellprogramm)?
 - Zellstate: Position, Orientierung, Form
 - Zellprogramm: Teil des Objekts soll nicht zu weit vom Objekt entfernt sein. Wenn zu weit weg, verschwindet es. Werden durch Differentialgleichungen erstellt.
- Was sind Sweeps in dem Zusammenhang?
 - Sweeps sind das Bewegen von Kurven im Raum, um Flächen zu erzeugen. In diesem Fall werden die Kurven von Objekt zu Zellen so bewegt, dass Sie ein 3D-Objekt erzeugen.

Knitwear

- Was ist das?
 - Simulation dünner 3D-Strukturen mit instanziierten Volumenelementen
- Wie ist eine Zelle aufgebaut?
 - Eine Zelle wird aus Teilfäden zusammen gewirbelt und die Teilfäden selbst bestehen aus Fasern, die auch verdreht werden.
- Welche Parameter kann eine Zelle haben?
 - **TODO**

Tensorproduktflächen

- Was sind Tensorproduktflächen? Wie werden sie erzeugt?
 - Wenn man eine Kurve mittels Sweep durch den Raum bewegt entsteht eine Fläche. Diese Fläche nennt man Tensorproduktfläche.
- Was sind die Eigenschaften von Tensorproduktflächen?

- **TODO**
- Was ist die Formel?
 - Man hat eine Kurve, bewegt die Kontrollpunkte und kombiniert beides, was einen Tensor ergibt

Subdivision Surfaces:

- Was ist das?
 - Ein Subdivision Surface ist eine glatte Fläche, die aus einem polygonalem Netz erzeugt wurde
- Wie geht man vor?
 - Man schneidet iterativ die Ecken weg wie beim Schnitzen, bis man eine abgerundete bzw. flache Fläche bekommt

Horizon Mapping

- Was ist das? Wie funktioniert es?
 - In einer weiteren Textur merkt man sich für jedes Texel das Höhenprofil
- Welche Informationen werden mit der Textur gespeichert?
 - Interpolierte Horizontwerte
 - Höhe
- Wie unterscheidet sich Horizon Mapping von Bump Mapping?
 - Man kann beim Horizon Mapping auch Schatten richtig darstellen

Was ist Parallax Mapping?

- Was ist das? Wie funktioniert es?
 - Eine Technik zur Verbesserung des Tiefeneindrucks bei Texturen
 - Verbesserung von Bump- oder Normalmapping
- Welche Informationen werden mit der Textur gespeichert?
 - Lokal Höheninformationen durch Höhenwerte vorgegeben
 - In welchem Winkel schneidet der View Ray das Polygon
- Was wird beim Parallax Mapping nachträglich korrigiert?
 - Bending Artefakte (mit bi- oder trilinearere Interpolation korrigiert)

Zusätzliche relevante Fragen:

Was ist der Dirac Impuls?

TODO

Erkläre den Painters Algorithm zum Abarbeiten von BSP-Trees.

TODO

Erkläre die Fourier-Transformation im Detail?

TODO

Was ist Tessellation?

- Ähnlich wie Subdivision Surfaces, nur dass man von einer Approximation ausgeht und die Normalvektoren der Approximation betrachtet, um sich einer Kurve anzunähern (da echte Kurven nicht möglich sind)

Wie ist eine parametrische Kurve definiert?

- Durch Parameter (z.B.) T , der zwischen 0 und 1 variiert. Bei Parameter 0 am Anfang der Kurve, bei 1 am Ende der Kurve.
- Jeder Punkt auf der parametrischen Kurve ist durch diesen Parameter durchwanderbar
- B-Spline und Bezier sind solche Kurven

Nützlich:

Vor- und Nachteile der einzelnen Datenstrukturen:

Kriterien: Rendern, Transformationen, Speicherbedarf, Kombinierbarkeit, Weiteres

Point clouds:

Menge von Punkten an Objektfläche

- Rendern: Sehr schnell (keine Rücksicht auf Nachbarschaft)
- Transformationen: Leicht
- Speicherbedarf: Hoch
- Kombinierbarkeit: schlecht
- Weiteres: Man braucht sehr viele Punkte für exakte Darstellung

Wireframe:

Objekt, das auch Knoten und Kanten besteht. Man hat eine Kantenliste. Die Eckliste enthält zwei Einträge für die Kanten. Beide Einträge sind Pointer zur Knotenliste, wo alle Knoten Informationen enthalten sind.

- Rendern: Schnell
- Transformationen: Leicht/schnell
- Speicherbedarf: Vermutlich auch hoch, nachdem Point clouds auch hoch sind(???)
- Kombinierbarkeit: eher schlecht, teilweise nicht möglich

- Weiteres: Kurven nicht darstellbar, Ungenau, keine Surfaces, keine Occlusion (Verdeckung)

Boundary Representation:

Ausweitung des Wireframemodells. Zusätzliche Facelist. Hierarchischer Aufbau. Nur Vertex list enthält tatsächliche Koordinaten. Reihenfolge der Verweise: Vertex-List <- Edge-List <- Face List

- Kombinierbarkeit: Nur Clipping mit einer Ebene easy, Vereinigung, Durchschnitt, Differenz schwer.
- Speicherbedarf: Hoch
- Weiteres: Durch Faces: Darstellung des Objekts als geschlossene Einheit.
- Restliche Kriterien gleich wie bei Wireframe, nur dass hier Occlusion darstellbar ist, da Faces vorhanden.

Winged Edge Data Structure:

Alternative zur B-Rep. Zentrales Element ist die edge, ansonsten enthält auch die winged edge data structure eine List für Vertices, Edges und Faces..

- Großer Vorteil im Vergleich zu B-Rep: Bei Winged Edge findet man benachbarte Polygone sehr schnell. Bei B-Rep muss man alle faces in der Liste durchgehen, um benachbarte zu finden.

Ab hier: Datenstrukturen, die Informationen zum gesamten Raum speichern, nicht nur zu Objekten im Raum.

BSP-Tree: Binary Space Partitioning Tree

Nimmt das Polygon eines Objekts im Raum, legt eine Eben durch und teilt den Raum in 2 Hälften. In Unterräumen wird wieder gesplittet. (Wurzelpolygon). Verdeckungsinformation wird so gespeichert.

- Rendern: Schnell
- Transformationen: Leicht/schnell, aber nur manche
- Speicherbedarf: Hoch
- Kombinierbarkeit: schnellere Kombination als bei B-Reps
- Weiteres: Kurven schwer darstellbar. Nur konvexe Polygone

Wie werden BSP-Trees abgearbeitet?

Painters Algorithm: Von hinten nach vorne (vom Wurzelpolygon aus) ausgeben. Dadurch stellt man sicher, dass man nie zwei Polygone aus verschiedenen Teilbäumen vergleichen muss.

- Transformationen: Easy, da affin
- Kombinierbarkeit: nur durch Umsortierung oder durch Umwandlung in B-Rep möglich.

kD Baum:

Sonderform des BSP-Baums. Trennebene unabhängig von Polygonen. Achsenparallele Trennung.

Octree:

Spezialform des kd-Baums, bei dem man den Raum immer in der Hälfte aller Achsen teilt -> 1 Teilwürfel. Können als Suchbäume verwendet werden.

Bintree:

Spezialfall des kD Baums. Ist ein 3D Baum, bei der die Teilungsreihenfolge immer xyzxyzxy... ist

Grid:

Raum unterteilt in regelmäßiges Gitter. Macht Kollisionsbehandlung einfach. Direkt adressierbare Gitterzellen.

