

VC Zusammenfassung

Contents

Visual Computing	4
Wahrnehmung.....	4
Auge.....	4
Stäpchen.....	4
Zäpfchen.....	5
Netzhautzellen.....	5
Informationsverarbeitung:	5
Aufmerksamkeit	5
Depth Cues	5
Erkennung	5
Bayes Decision History	5
Sliding Window.....	5
Appearance Model	5
Erkennungsarten	6
Fourier-Theorie.....	6
Dirichlet Bedingungen	6
Charakterisitk Fourier-Wellen	6
Gerade/ Ungerade Funktion	6
Abtastung von Signalen.....	6
Bilder	6
Histogramm	6
Aliasing	6
Pixeloperationen	7
Frequenzraumfilter	7
Filterung	7
Kompression.....	7
Bildverarbeitung.....	7
Image Blurring	7
Wiener-Filter	8
Scale-Space-Ansatz.....	8
Mehrschrittverfahren.....	8
Grafikpipeline	8
Räumliche Datenstrukturen	9

Hüllkörperhierarchie	9
Raumteilung Regulär: Gitter.....	9
Raumteilung Irregulär / Hierarchisch:	9
Schattierungsverfahren.....	9
Clipping.....	9
Culling.....	9
Rasterisierung.....	10
Zerlegen von Primitiven (Linien, Polygone) in Pixel (Scanline)	10
Algorithmus von Bresenham: $\max dX, dY + 1$	10
Z-Buffer Algorithmus (Verdeckungsrechnung).....	10
Painters Algorithmus (Überlappung/ Anordnung)	10
Transformationen.....	11
Transformationen in Grafikpipeline	11
Transformationen.....	11
Projektionen	11
3D-Visualisierung.....	11
Voronoi-Diagramm	11
Delaunay-Triangulierung	12
Volumenvisualisierung	12
Marching Squares.....	13
Culling	13
Volumen-Rendering-Pipeline	13
Informationsvisualisierung	14
Card-Pipeline	14
Techniken	14
Slice-and-dice Algorithmus.....	15
Parallele Koordinaten	15
Farbe.....	15
Technische Farbräume	15
Farbeigenschaften	15
Beobachtermetametrie	16
CIELAB.....	16
Farbwahrnehmungsmodelle	16
Simultankontrast	16
Effekte	16
User Interfaces	16

Interaktionsmöglichkeiten	16
Metriken	16
Query-Modalitäten.....	16

Visual Computing

Visual Computing ist eine Kombination mehrerer Bereiche der Informatik in denen im Wesentlichen mit Bildern und Modellen gearbeitet wird.

Die Fachgebiete Computergraphik und Computer Vision dürfen nicht länger voneinander getrennt betrachtet werden. Die Schnittstelle beider Gebiete verdient besondere Beachtung.

Themenbereiche:

- 3D Internet
- Skalierbare Objektmodellierung und -erkennung
- Big Data / Visual Analytics
- Scene Understanding, Verstehen und Analysieren von 3D/4D Szenen

3D oder generalisierte Dokumente: Funktionalität der Behandlung textueller Dokumente übertragen auf den verallgemeinerten Dokumentbegriff.

Retro-Digitalisierung: Entwicklung von effektiven Massendigitalisierungs-Werkzeugen und Workflows, Community-based

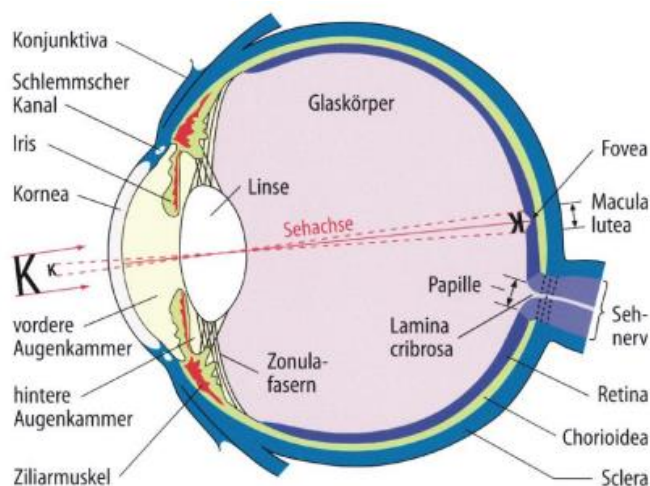
Digital CreATION. Shape-based Reconstruction

Wahrnehmung

Auge

Das Auge besteht aus der Hornhaut (Kornea), Linse (zur Scharfstellung), Iris (Blendenmechanismus), Retina (Netzhaut)

- Fovea Centralis (Gelber Fleck): Bereich mit höchsten Auflösung
- Blinder Fleck: Hier geht der Sehnerv ab



Stäbchen

- Außerhalb der Fovea
- Empfindlichkeitsmaximum bei 498 nm (grün)
- Lichtempfindlich für skoptisches sehen (Nachtsehen)
- Mehr als Zäpfchen

Zäpfchen

- Innerhalb der Fouvea
- Empfindlichkeitsmaximum je nach Typ bei rot, grün, blau
- Photopisches sehen (Tagsehen)

Netzhautzellen

- Horizontalzellen: kombinieren mehrere Rezeptoren einer Region
- Bipolarzellen: Informationsfilter
- Amakrinzellen: zeitliche Verarbeitung
- Ganglienzellen: integrieren Informationen

Informationsverarbeitung:

Perception (sensory) -> decision (cognition) -> response (motor)

Aufmerksamkeit

- Filter im Gehirn
- Wenn einem große Änderungen nicht auffallen, nennt man das Change Blindness

Typen:

- Gewählt: zwischen mehreren Möglichkeiten wird eine gewählt, die fokussiert werden soll
- Geteilt: Multitasking
- Erfasst: ein Reiz zieht alle Aufmerksamkeit auf sich
- ➔ Visuelle Effekte für frühe Wahrnehmung: Farbe, Form

Depth Cues

1. Binokulare Depth Cues (zwei Augen): Disparität /Parallaxe, Konvergenz, Akkomodation
2. Pictorial Depth Cues (ein Auge): Linearperspektive, Verdeckung, Fokus /Blur, Schattenwurf...
3. Dynamic Depth Cues: Bewegungsparallaxe, Interposition, kinetische Tiefeneffekte...

Erkennung

Bayes Decision History

- Priors: a priori Wahrscheinlichkeit $P(C_k)$
- Bed. Wahrscheinlichkeit: $P(x|C_k)$
- Posteriors: Wahrscheinlichkeit der Klasse C_k gegeben Merkmalsvektor x

$$Posterior = \frac{Likelihood \cdot Prior}{Normalization Factor}$$

- Likelihood: Wahrscheinlichkeit, dass Klasse C_k Merkmal x aufweist
- Naive Bayes Klassifikator: Annahme, dass Merkmale statistisch unabhängig sind

Sliding Window

Verfahren der Gesichtsdeduktion. Das Eingabebild wird in Ein-Pixelschritten horizontal & vertikal gescannt. Das Bild wird um den Faktor 1,2 verkleinert & der Vorgang wird wiederholt, bis das Bild zu klein ist.

Appearance Model

1. Repräsentation des Objektes: lokale Merkmale, globale Anordnung der Merkmale
2. Trainingsdaten: positive Beispiele, negative Beispiele
3. Klassifikator & Lernmethode

Erkennungsarten

- Verifikation: gehört Person zum Referenzmerkmalsdatensatz
- Identifikation: Vergleich mit RMD & User-ID wird ermittelt

Fourier-Theorie

Dirichlet Bedingungen

Jede Funktion, die die Dirichlet-Bedingungen erfüllt:

1. Die Anzahl der Unstetigkeiten innerhalb einer Periode ist endlich
2. Die Anzahl der Maxima und Minima innerhalb einer Periode ist endlich
3. Die Funktion ist in jeder Periode integrierbar

Kann durch eine Summe von Kosinus- und Sinusfunktionen dargestellt werden.

Charakteristik Fourier-Wellen

Ortsbereich: An jeder Stelle gibt sich der Funktionswert durch die Überlagerung aller Wellen -> jede Welle ist überall im Bild aktiv

Frequenzbereich: Alle Werte der Funktion $f(x)$ werden bei der Berechnung von dem jeweiligen $F(u)$ einbezogen -> alle Funktionswerte werden bei der Berechnung jeder Welle berücksichtigt

- Jede Funktion, die die Dirichlet-Bedingungen erfüllt, lässt sich als Summe von Sinus- & Cosinuswellen darstellen
- Eine Faltung zweier Funktionen

Gerade/ Ungerade Funktion

Gerade Funktion: $f(-t) = f(t)$, alle $b_n = 0$

Ungerade Funktion: $f(-t) = -f(t)$, alle $a_n = 0$

$$f(x) = \sum_{n=0}^{\infty} (a_n \cdot \cos(n \cdot x) + b_n \cdot \sin(n \cdot x))$$

➔ Fourierdarstellung: Zerlegen einer Frequenz in ihre Frequenzbestandteile

Abtastung von Signalen

Abtastsignal: Diskretisierung einer kontinuierlichen Funktion mit einer Kammfunktion

Abtasttheorem von Whittaker Shannon: Abtastfrequenz $\frac{1}{\Delta x}$ muss mindestens doppelt so hoch wie U_G sein, damit das Signal fehlerfrei rekonstruierbar ist -> $\frac{1}{\Delta x} > 2U_G$

Bilder

Histogramm

- Bilddynamik: Bereich reeller Lichtintensitäten, der auf die Grauwertskala abgebildet wird
- Bildhelligkeit: dunkles Bild = niedrige Werte, helles Bild = hohe Werte
- Bildkontrast: hoher Kontrast = gestreute Werte, niedriger Kontrast = enge Werte

Aliasing

Frequenzen, die oberhalb der halben Samplingfrequenz liegen, werden als niedrige Frequenzen interpretiert, da eine vollständige Rekonstruktion des Ausgangssignals nicht möglich ist

Pixeloperationen

- Bildnegativ: $g[m, n] = f_{max} - f[m, n]$
- Binärisierung/ Thresholding: $g[m, n] = \begin{cases} f_{max}, & \text{falls } f[m, n] > \tau \\ f_{min}, & \text{falls } f[m, n] \leq \tau \end{cases}$
- Grauwertfensterung: Hervorheben eines bestimmten Intensitätsintervalls im Bild
- Kontrastspreizung: Abbildung der Grauwerte auf eine neue Grauwertskala anhand einer einwertigen, monotonen Funktion
- Histogrammausgleich: Transformation der Grauwertskala anhand der Kurve der Summenwahrscheinlichkeit: $p(g) = \max(\text{Intensität}) \cdot \sum_{i=0}^g p(i)$
- Mittelung: Unterdrückung von unkorreliertem Rauschen durch Mittelung über k Aufnahmen

$$g(m, n) = \frac{1}{k} \sum_{i=0}^{k-1} f_i(m, n)$$

Frequenzraumfilter

- Vorteile: schnelle Berechnung, einfache Handhabung
- Nachteil: Approximation der Spezifikation aus dem Frequenzbereich

Filterung

Lineare Filterung:

- Tiefpass-Filter
 - Koeffizienten positiv + normalisiert $\sum = 1$
 - Produziert nur positive Werte
 - Randeffekte
 - Weichzeichnungsfilter: Mittelwertfilter, Gaussian Filter, entfernt/ reduziert Rauschen, verringert störendes Druckraster
- Hochpass-Filter
 - Koeffizienten pos + neg -> erste Ableitung/ Differenzierbar
 - Koeffizienten normalisiert -> zweite Ableitung
 - Produziert pos + neg Werte

➔ Kantenextraktion

Kompression

JPEG:

1. Umwandlung in den $Y C_R C_B$ -Farbraum
2. Farb-Subsampling
3. Diskrete Kosinustransformation
4. Quantifizierung
5. Kodierung der Koeffizienten

Eliminierung redundanter Daten: Kodierung, Nachbarschaftsbeziehungen, Psychovisuelle Eindrücke

Bildverarbeitung

Image Blurring

Image Blurring beschreibt das Weichzeichnen eines Bildes, wodurch es verschwommen wirkt und Details verloren gehen. Beispiel: Gaussfilter / Boxfilter

Probleme:

1. Blurring-Kernel kann unendlich klein werden, sodass es beinahe zu einer Division durch 0 kommt
2. Es gibt immer Rauschen: $G = a(t) + n$

Wiener-Filter

- ➔ R = Verhältnis Rauschen zu Signal
- Großes R : Tiefpass-Filter -> behält grobe Struktur, verwischt Kanten, entfernt Rauschen
- Kleines R : Hochpass-Filter -> entfernt grobe Struktur & Kanten, verstärkt Rauschen
- Optimales R : Bandpass-Filter -> entfernt Rauschen, behält grobe Struktur, verstärkt Kanten-Struktur leicht (Deblurring)

Vorteile:

- Schnell
- Häufig verwendet
- Leicht zu implementieren

Nachteile:

- Nur ein Filter für Bild
- Keine lokalen/ spezifischen Verbesserungen
- Ein Wert für R

Scale-Space-Ansatz

Entfernt Image-Deblurring, hinzufügen von vielen Termen kann Rauschen wieder Verstärken

Mehrschrittverfahren

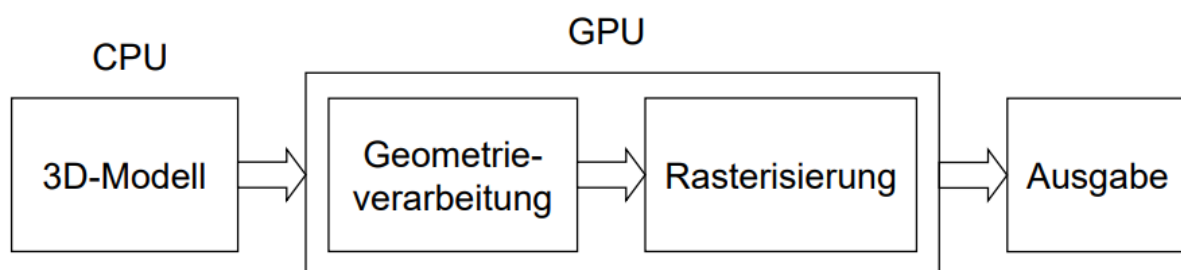
Perona-Malik: Stoppzeit, Parameter k bestimmt, welcher Gradient erhalten bleiben soll, während Rest vermischt wird

Kleines k : Erhält fast alle Gradienten (starke und schwache Kanten + Rauschen)

Großes k : erhält nur große Gradienten (starke Kanten)

- ➔ Total Variation: Distance Penalty
- ➔ Entfernen/ verwischen Rauschen & verstärken Kanten

Grafikpipeline



Uncanny Valley: Eine hypothetische Beziehung zwischen dem Grad der Ähnlichkeit eines Objekts mit einem Menschen und der emotionalen Reaktion auf ein solches Objekt.

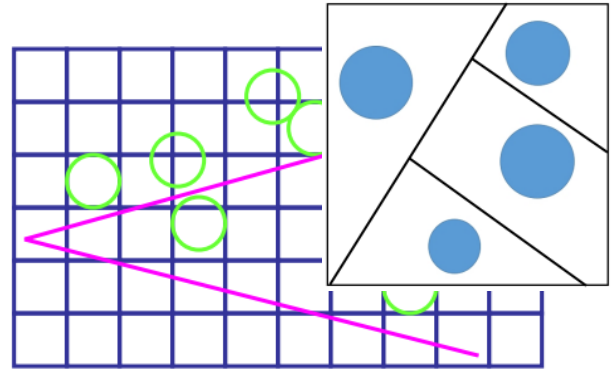
Räumliche Datenstrukturen

Hüllkörperhierarchie

Hüllkörper müssen einfach sein, d.h. Schnitttests mit anderen Primitiven (Sichtvolumen, Sehstrahl) müssen sich einfach berechnen lassen

Raumteilung Regulär: Gitter

Einfach, Objekt ist in mehreren Zellen enthalten, kann sich der Geometrie nicht anpassen, Speicheraufwändig, effizient traversierbar, schneller Zugriff auf Nachbarn, Voxel- / Volumendarstellung



Raumteilung Irregulär / Hierarchisch:

k-d Tree (achsenparallele BSP-Bäume)

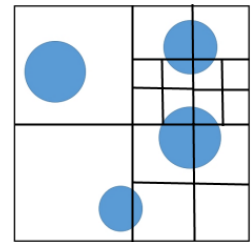
Teilen wie BSP aber nur horizontal und Vertikal

Uniform: Quadtree (2D) / Octree (3D)

Raumteilung jedes mal, falls mehr als ein Objekt in einer Zelle

BSP (Binary Space Partitioning) Tree

Raum wird binär unterteilt: Jeder Knoten entspricht einer Unterteilungsebene, welche den Raum in zwei Halbräume unterteilt, man teilt an den durch die Polygone induzierten Ebenen



K-d-Tree

Quadtree

Schattierungsverfahren

- Phong-Shading $\rightarrow I_{total} = I_{amb} + I_{diff} + I_{spec}$
- Gouraud-Shading
- Flatshading

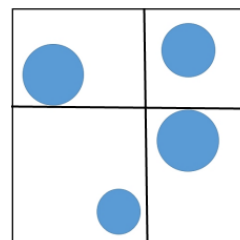
BSP Tree

Clipping

Abschneiden von Objekten am Rand eines gewünschten Bildschirmausschnitts

Culling

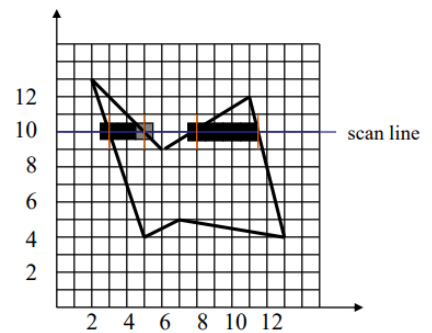
Verdeckungsrechnung im Objektraum, Rückseiten werden nicht berechnet



Rasterisierung

Zerlegen von Primitiven (Linien, Polygone) in Pixel (Scanline)

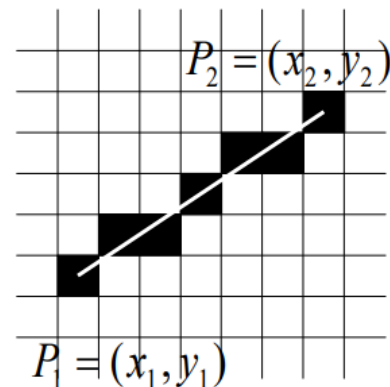
Finde Schnittpunkt der scan line mit allen Kanten des Polygons
 Sortiere Schnittpunkte nach wachsender x-Koordinate
 Fülle Pixel zwischen Paaren aufeinanderfolgender Schnittpunkte
 Regel von der ungeraden Parität
 Parität ist am Anfang 0
 Mit jedem Schnittpunkt um eins inkrementiert
 Pixel wird gesetzt falls Parität ungerade



Algorithmus von Bresenham: $\max(dx, dy) + 1$

```

dx = xend-xstart, dy = yend-ystart
x = xstart
y = ystart
Pixel (x,y) markieren
fehler = dx/2
WHILE x < xend
  x = x + 1
  fehler = fehler - dy
  IF fehler < 0 THEN
    y = y + 1
    fehler = fehler + dx
  Pixel (x,y) markieren
  
```



Achtung: im ersten Oktanten muss $0 < dy \leq dx$ sein, sonst vertauschen

Z-Buffer Algorithmus (Verdeckungsrechnung)

Initialisierung: Bildspeicher: Hintergrundfarbe, z-Speicher: maximaler z-Wert

Für jeden Punkt (x,y) eines darzustellenden Polygons

1. Berechne $z(x,y)$: Perspektivische transformation erlaubt keine lineare Interpolation mehr!
2. Ist $z(x,y)$ kleiner als der untere (x,y) bereits gespeicherte Wert: Schreibe $z(x,y)$ in den z-Speicher und den zugehörigen Farbwert an der Stelle (x,y) in den Bildspeicher

Vorteile: Jede Szene kann behandelt werden, Komplexität ist unabhängig von der Tiefenkomplexität, leicht zu realisieren, Objekte können nachträglich eingefügt werden

Nachteile: Pro Bildpunkt wird nur ein Objekt gespeichert -> Abtastfehler, keine Transparenz, Genauigkeit beschränkt

Painters Algorithmus (Überlappung/ Anordnung)

Tiefe: z-Wert, $z \in [z_{min}, z_{max}]$, das am weitesten entfernte zuerst (z_{max})

Sortiere Polygone nach z-Wert
 Falls z-Intervalle überlappen müssen Schnittpolygone berechnet werden
 Beginne das Zeichnen mit dem Polygon mit größtem z-Wert

Transformationen

Transformationen in Grafikpipeline

1. Modelling Transformations -> ordne 3D Objekte im Raum an
2. Viewing Transformations -> wähle Betrachtungsstandort
3. Project Transformations -> projiziere Viewing Volume in 2D
4. Viewport Transformations -> wandle Bildschirmkoordinaten um

Transformationen

Affine Abbildungen sind nicht kommutativ, bei einer Skalierung sind alle Werte außerhalb der Diagonalen gleich null, Projektionstransformationen sind keine affinen Abbildungen

Translation: jeder Wert wird um den gleichen Vektor d verschoben

Skalierung:

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} s_1 & 0 & 0 & 0 \\ 0 & s_2 & 0 & 0 \\ 0 & 0 & s_3 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Scherung:

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & s_2 & s_5 & 0 \\ s_1 & 1 & s_6 & 0 \\ s_3 & s_4 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Rotation:

1. Rotationszentrum in Ursprung verschieben
2. Rotation durchführen
3. Zurückverschieben des Rotationszentrum

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & \sin(\alpha) & 0 & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Projektionen

Perspektivische Projektion: realistischer, Winkel und Längenverhältnis ändern sich, parallele Geraden bleiben nicht parallel

Parallele Projektion: weniger Realismus, Winkel ändert sich nicht, parallele Geraden bleiben parallel

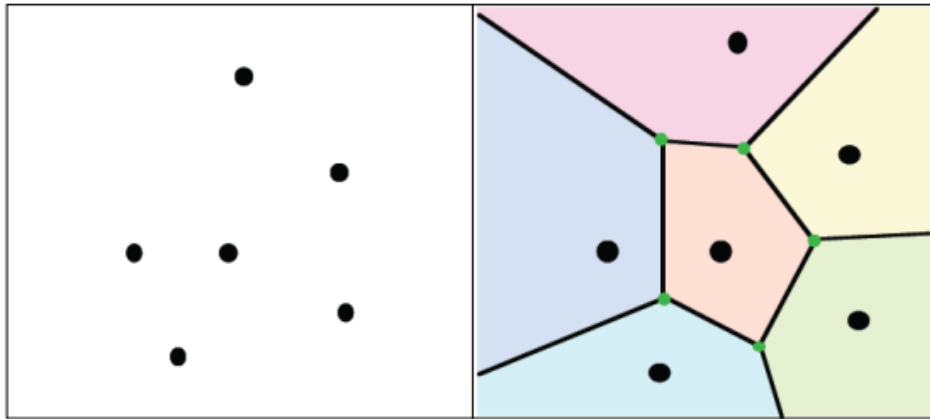
3D-Visualisierung

Voronoi-Diagramm

Für jeden (projizierten) Punkt s_i kann eine Voronoi-Zelle definiert werden:

- Eine solche enthält alle Punkte, die näher an s_i als zu allen anderen Orten liegen
- Kanten einer Voronoi-Zelle -> Punkte mit zwei nächsten Punkten
- Knoten einer Voronoi-Zelle -> Punkte mit drei oder mehr nächsten Punkten

Die Voronoi-Zellen „parkettieren“ die 2D-Fläche



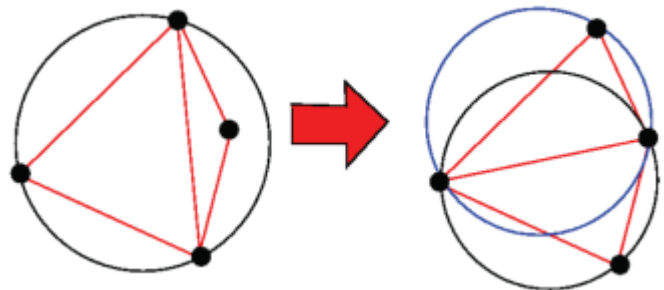
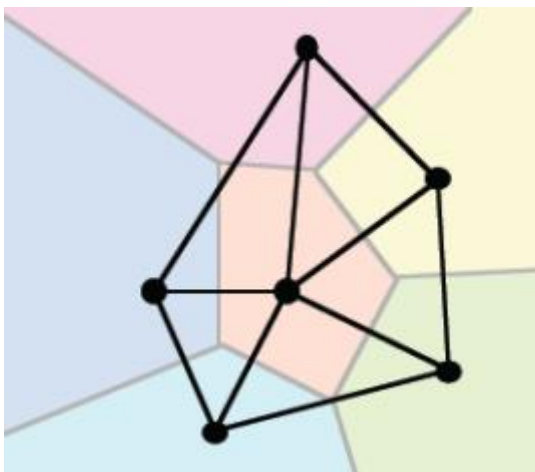
Delaunay-Triangulierung

Delaunay-Triangulation: Die Vertices benachbarter Zellen im Voronoi-Diagramm werden mit Linien verbunden. Das muss so passieren, dass der Umkreis eines entstandenen Polygons keine weiteren Punkte enthält

Von der Mitte der Kanten immer orthogonale Linien zeichnen, die die Flächen des Voronoi-Diagramms abgrenzen

Dreiecksnetz in dem alle Umkreise von allen Dreiecken im Netz leer sind

Edge-Flipping wird zur Korrektur angewendet, wenn sich im Umkreis eines Dreiecknetzes noch mindestens ein weiterer Punkt befindet



Volumenvisualisierung

Metadarstellung:

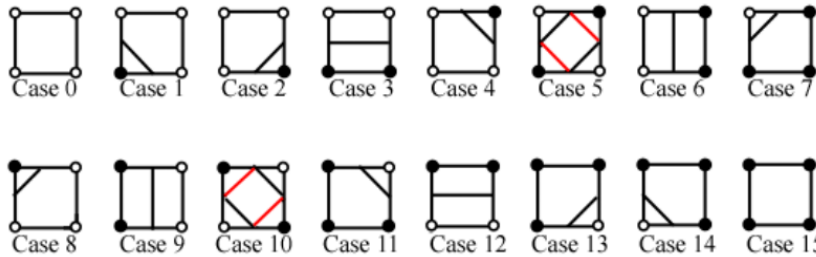
- Indirekt: Visualisierung mit Generierung einer Zwischendarstellung
- Direkt: ohne

Komplexität

- Indirekt: abhängig von Polygonen
- Direkt: abhängig von Anzahl der Voxel + Auflösung der Anzeigefläche

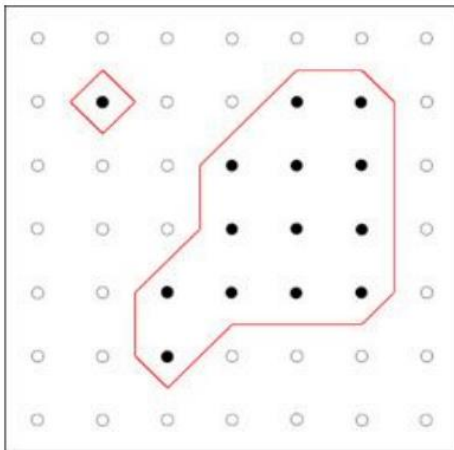
Marching Squares

Bildzellen seien durch ihre vier umgebenen Pixel definiert
Pixel werden betrachtet, wenn sie größer oder gleich des isowertes sind
16 mögliche Kombinationen:



Beschränkung auf 5 Fälle, wenn Rotationen und Invertierungen der Pixel-Status berücksichtigt werden

Annahme: Eine Kontur passiert eine Zellengrenze zwischen zwei benachbarten Pixeln genau ein mal -> Generierung einer Kontur, die die Zellengrenzen kreuzt

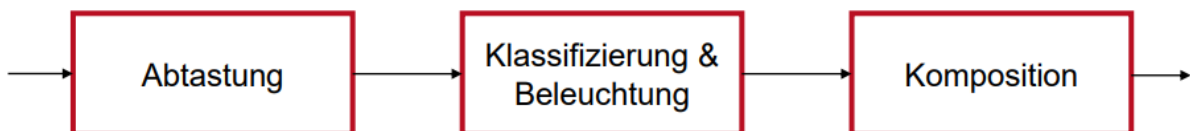


Marching Cubes -> 3D-Variante von Marching Squares

Culling

- Backface-Culling: Zum Betrachter gerichtete Rückseiten nicht gezeichnet
- View-Frustum-Culling: Polygone, die sich ganz oder teilweise außerhalb des View-Frustums befinden, werden nicht gezeichnet
- Occlusion-Culling: Polygone nach Tiefe sortiert und nur gerendert, wenn sie nicht vollständig durch andere verdeckt werden

Volumen-Rendering-Pipeline



1. Abtastung (Sampling)
2. Klassifizierung & Beleuchtung
3. Komposition: Back-to-front, Front-to-back

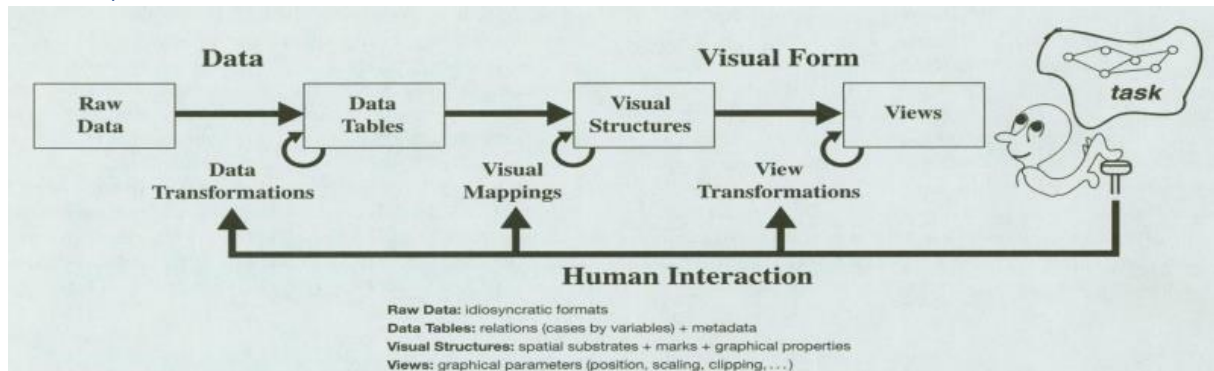
Transferfunktionen werden verwendet, um Skalaren Werten in Volumendaten optische Eigenschaften wie Transparenz oder Farbe zuweisen

Meshreduktion: Bei der Meshreduktion wird die Anzahl der Polygone verringert, wobei die „Größe“ der Vereinfachung stark vom Szenario abhängt (Genauigkeit vs. Zeitbegrenzung).

Meshglättung: Das Ziel der Mesh-Glättung ist die Bereitstellung einer guten Visualisierung sowie der Artefakt-Reduzierung und Entfernung von „Löchern“. Die Herausforderung hierbei ist das Volumen zu erhalten.

Informationsvisualisierung

Card-Pipeline



Techniken

1D:

- Kuchendiagramm
- Balkendiagramm

Zeitreihen:

- Liniengrafik

2D/3D:

- Scatterplot

3D/nD:

- Scatterplotmatrix
- Parallele Koordination

Hierarchien/ Bäume:

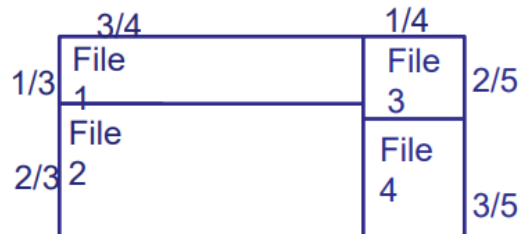
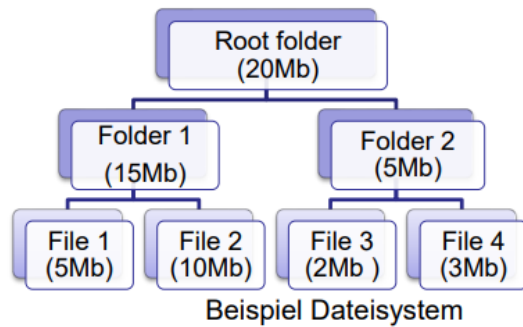
- Node-Link-Diagramm
- Treemap

Netzwerk/Graphen:

- Node-Link-Diagramm

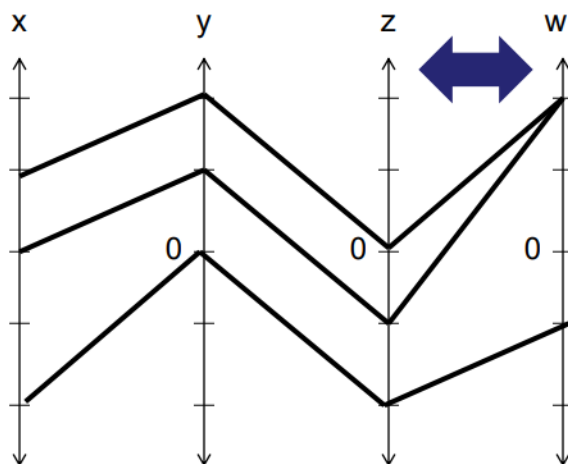
Slice-and-dice Algorithmus

Rekursive Aufteilung eines Rechtecks anhand der Baumstruktur
 Beginnend mit der Wurzel
 Teile anhand der Teilbaumgrößen
 Alterniere zwischen horizontaler und vertikaler Aufteilung



Parallele Koordinaten

Idee: Koordinaten parallel zueinander
 Aus Punkt im Scatterplot wird eine Linie in parallelen Koordianten
 Mehrere Dimensionen nebeneinander abgebildet
 Achsenanordnung ist wichtig, laufen mehrere Linien aufeinander zu oder Teilen sich, sollten diese am Anfang oder am Ende stehen



Farbe

Technische Farbräume

Geräte RGB, Geräteunabhängige RGB, YcbCr, HSI (HSV, HSL), CMY /CMYK

Farbeigenschaften

1. Helligkeit
2. Relative Helligkeit = $\frac{\text{Helligkeit}}{\text{Helligkeit(Weiß)}}$
3. Farbton
4. Farbigkeit
5. Buntheit = $\frac{\text{Farbigkeit}}{\text{Helligkeit(Weiß)}}$

$$6. \text{ Sättigung} = \frac{\text{Farbigkeit}}{\text{Helligkeit}} = \frac{\text{Bundheit}}{\text{Relative Helligkeit}}$$

Beobachtermetametrie

Zwei Farbreize erzeugen bei gleichen Beleuchtungsbedingungen für eine Person die gleichen, für eine andere unterschiedliche Farbvalenzen

CIELAB

Gegenfarbraum, nahezu Wahrnehmungsgleichabständig

Farbwahrnehmungsmodelle

Farbwahrnehmungsmodelle ermöglichen eine Anpassung der Farbreize für den Farbabgleich bei unterschiedlichen Betrachtungsbedingungen

Simultankontrast

HG auf dem ein Farbreiz präsentiert wird, beeinflusst die wahrgenommene Farbe -> Farbverschiebung folgt Gegenfarbtheorie

Effekte

Crispening-Effekt: Der wahrgenommene Farbunterschied zweier Farbreize wird durch einen ähnlichen Hintergrund erhöht

Stevens-Effekt: Kontrast steigt mit der Leuchtdichte

Hunt-Effekt: Farbigkeit steigt mit der Leuchtdichte

User Interfaces

Interaktionsmöglichkeiten

1. Kommandozeile
2. Menüs
3. Formulare
4. Fragen & Antworten
5. Direkte Manipulation
6. 3D-Umgebung
7. Natürliche Sprache
8. Gesten

➔ 3D-Interaktion: Mehrdeutigkeit bei der Art der Bewegung > unendlich viele Cursorpositionen

➔ WIMP: Windows, Icons, Menus, Pointers

Metriken

- Nicht-Negativität
- Definitheit
- Symmetrie
- Dreiecksungleichung

Query-Modalitäten

- Query-by-Text
 - Query-by-Sketch
 - Query-by-Example
 - Explorative Suche: keine konkrete Suchanforderungen
- ➔ Inhalt Multimedia beschreiben: textueller Deskriptor

➔ Generalisierte Dokumente: Sind in der Datenbank dokumentiert