Direkte Schattierungsverfahren (Gouraud)

Kontinuierliche Schattierung von Polygonen (Gouraud-Shading)

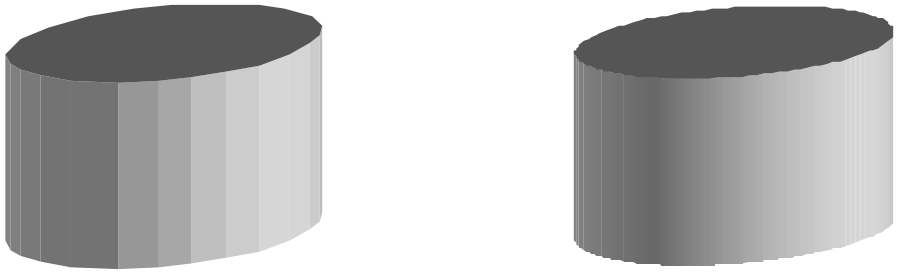
bisher:

- Zerlegung gekrümmter Flächen (Zylinder, Kugeln, B-Spline-Flächen etc.) in Polygone (meist Drei- oder Vierecke)
- Einzelne Schattierung der Polygone gemäß ihrem Normalenvektor (Flat Shading)
- Mangelhafte Realitätstreue der Flat-Shading-Methode aufgrund störend wirkender Diskontinuitäten an den Polygongrenzen

→ Gouraud-Shading [H. Gouraud 1971] beseitigt die Diskontinuitäten

- durch Richtungsvektoren in den Vertices (statt nur eines Richtungsvektors für das ganze Polygon beim Flat-Shading) und
- *Interpolation der Schattierung* (Intensitäten) im Inneren der Polygone.

Direkte Schattierungsverfahren (Gouraud)



Gegenüberstellung von Flat-Shading und Gouraud-Shading

Flat-Shading:

- Beleuchtungsberechnung mit nur einer Flächennormale pro Polygon

Gouraud-Shading in der Übersicht:

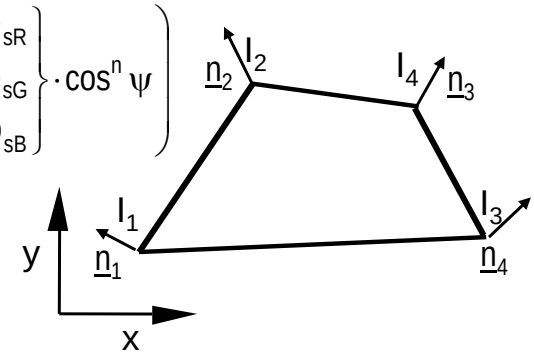
- Es werden „Normalenvektoren“ n_i für jeden Eckpunkt P_i des Polygons ermittelt.
- Die Herleitung der „Normalenvektoren“ n_i ist aus der Originaloberfläche (z. B. Zylinder, Kegel, Bèzier-Fläche) oder Nachbarpolygonen möglich.
- Für jeden Eckpunkt: Berechnung der Beleuchtungsintensität I_i (z. B. nach dem Phong-Beleuchtungsmodell).

Direkte Schattierungsverfahren (Gouraud)

- Normalen $\underline{n}_i$ der Eckpunkte werden entweder direkt aus den Flächen (z.B. Regelgeometrien, bei Kugel z.B. Richtung des Radiusvektors) oder aus den Flächennormalen der benachbarten Polygone durch flächengewichtete Mittelung berechnet.
- Die Schattierungsrechnung erfolgt für die Eckpunkte und liefert die I_i. Zur Erinnerung:

$$\begin{Bmatrix} I_R \\ I_G \\ I_B \end{Bmatrix} = \begin{Bmatrix} I_{aR} \\ I_{aG} \\ I_{aB} \end{Bmatrix} \cdot k_a \cdot \begin{Bmatrix} O_{aR} \\ O_{aG} \\ O_{aB} \end{Bmatrix} + f_{att} \cdot \begin{Bmatrix} I_{pR} \\ I_{pG} \\ I_{pB} \end{Bmatrix} \cdot \left(k_d \cdot \begin{Bmatrix} O_{dR} \\ O_{dG} \\ O_{dB} \end{Bmatrix} \cdot \cos\alpha + k_s \cdot \begin{Bmatrix} O_{sR} \\ O_{sG} \\ O_{sB} \end{Bmatrix} \cdot \cos^n \psi \right)$$

$$\cos\alpha_i = \underline{V}_l^T \cdot \underline{n}_i \,, \qquad \cos^n \psi = \left(\underline{V}_r^T \cdot \underline{V}_e \right)^n$$



$\underline{V}_l$: Einheitsvektor in Richtung Lichtquelle,

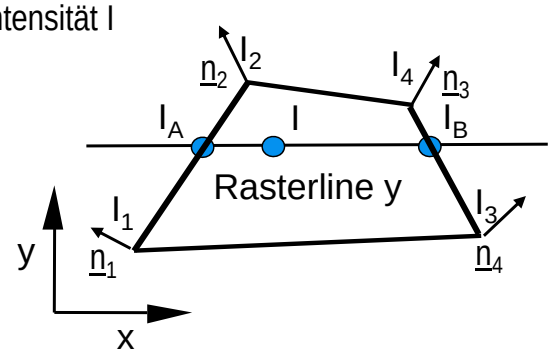
$\underline{V}_r$: Einheitsvektor in Reflexionsrichtung, in der Ebene $\underline{n}_i \times \underline{V}_l \rightarrow -\alpha_i$ gegen $\underline{n}_i$

$\underline{V}_e$: Einheitsvektor in Kamerarichtung (Projektionszentrum)

Direkte Schattierungsverfahren (Gouraud)

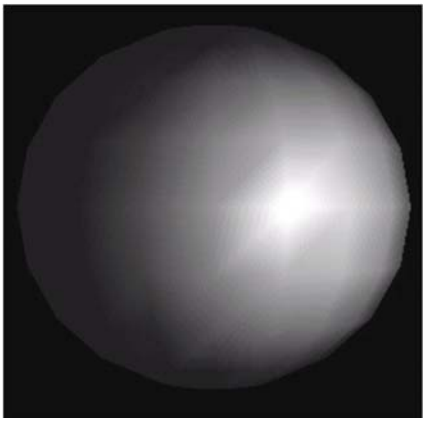
- Normalen $\underline{n}_i$ der Eckpunkte werden entweder direkt aus den Flächen (z.B. Regelgeometrien, bei Kugel z.B. Richtung des Radiusvektors) oder aus den Flächennormalen der benachbarten Polygone durch flächengewichtete Mittelung berechnet.
- Die Schattierungsrechnung erfolgt für die Eckpunkte und liefert die I_i.
- Bei der Rasterkonvertierung wird zwischen den Eckwerte I_i linear interpoliert (zwischen I_A und I_B) und damit die Intensität jedes Pixels der Rasterlinie berechnet (Intensität I steht hier für die Leuchtdichte oder für Farbwerte usw.)

Die Interpolation erfolgt nach dem gleichen arithmetischen Muster wie die Interpolation der z-Werte im z-Puffer-Verfahren oder wie beim Zeichnen von Geraden (d. h. <u>inkrementell, mit Ganzzahlarithmetik, Bresenham</u>).



Direkte Schattierungsverfahren (Gouraud)

Resultat: Kontinuierlich schattierte dreidimensionale Oberflächen



Gouraud-schattierte Kugeln (links 200, rechts 1000 Dreiecke)

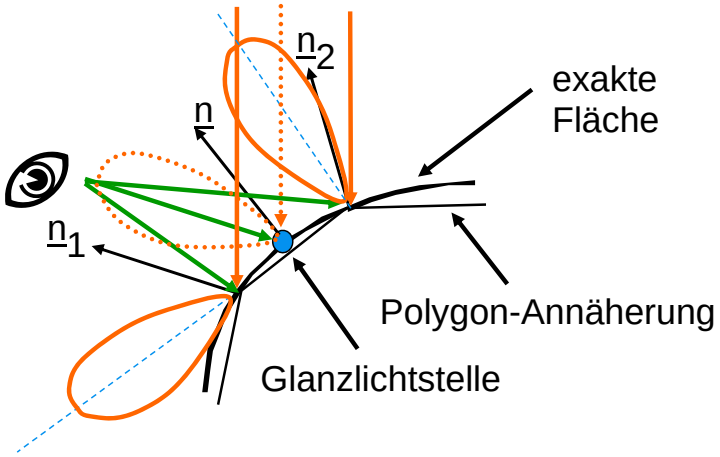
Nachteile des Gouraud-Shading, bedingt durch die lineare Interpolation:

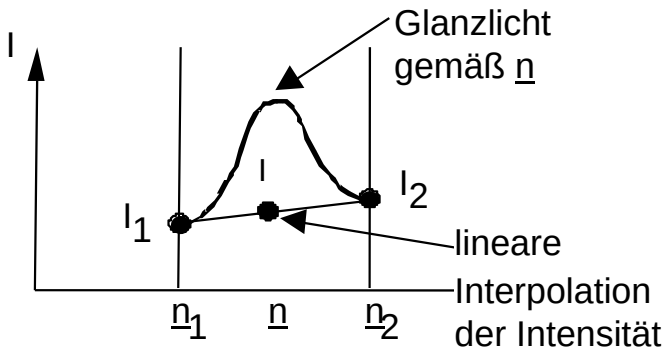
- *Fehlen von gut ausgeprägten Glanzlichtern*
- *Mach-Band-Effekt*: Kontrastverstärkung durch das Auge an den Übergängen zwischen Polygonen

Direkte Schattierungsverfahren (Gouraud: fehlende Glanzlichter)

Fehlende Glanzlichter

Auf Grund der linearen *Interpolation von Intensitäten können Glanzlichter*, die auf spekulare Reflexion zurückzuführen sind, *verloren gehen oder abgeschwächt / verschmiert werden*.
Das wird umso kritischer, je spitzer die spekulare Reflexion ist (großes n im $\cos^n$ - Term).

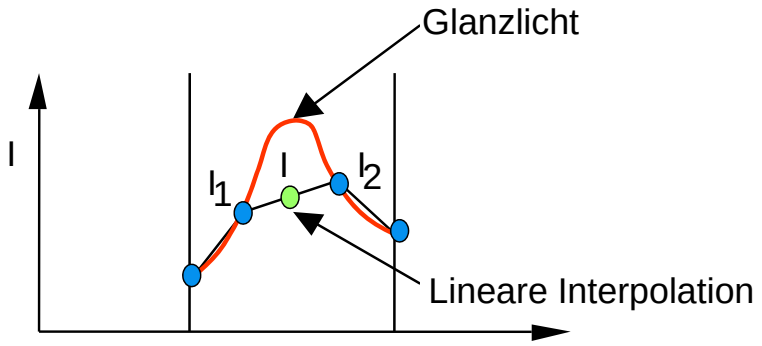




Annäherungsfehler beim Darstellen von Glanzlichtern

Direkte Schattierungsverfahren (Gouraud: fehlende Glanzlichter)

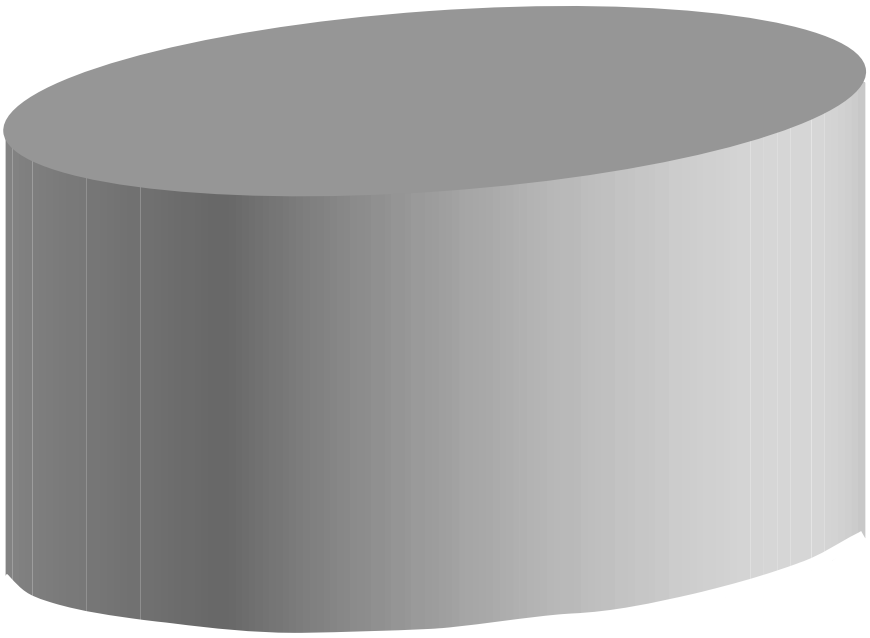
Feinere Unterteilung der Oberfläche verbessert Resultat:



Verbesserung des Gouraud-Schattierungsverfahrens durch feinere Polygonunterteilung

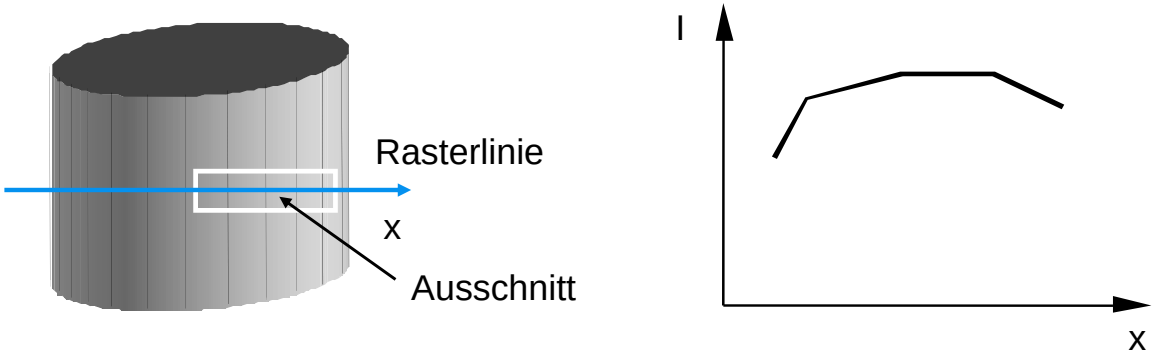
Direkte Schattierungsverfahren (Gouraud: Mach-Band-Effekt)

Der Mach-Band-Effekt



Direkte Schattierungsverfahren (Gouraud: Mach-Band-Effekt)

Der Mach-Band-Effekt



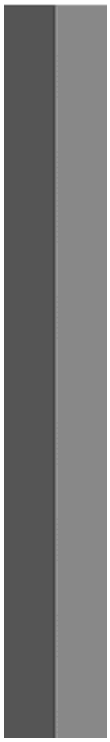
Die lineare Interpolation der Leuchtdichte zwischen den Polygonkanten entlang der Rasterlinie führt zu einem Verlauf, der durch plötzliche Änderungen im Anstieg der Intensität gekennzeichnet ist (nicht stetig differenzierbar)

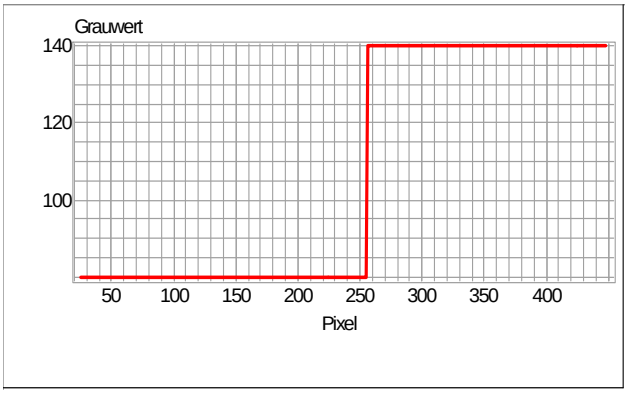
Direkte Schattierungsverfahren (Gouraud: Mach-Band-Effekt)

Der Mach-Band-Effekt: ein physiologisches Phänomen (Ernst Mach, 1865)

- *Gleiche Information benachbarter Rezeptoren* wirkt bei der weiteren visuellen Verarbeitung *lateral inhibitorisch* (hemmend) auf die lokale Lichtempfindung.
- Modellhaft entstehen neben dem eigentlichen Helleindruck *auch „Signale“*, die dem *Helligkeitsgradienten* (erste Ableitung) und dem *Laplacefilter-Output* (Laplacian of Gaussian LoG oder auch Marr-Hildreth-Filter, zweite Ableitung) entsprechen.
- Die Empfindung wird insgesamt nicht nur durch die Lichtintensität selbst, sondern auch durch die Überlagerung mit ihrer ersten und zweiten räumlichen Ableitung bestimmt.
- Das führt zu einer *Verstärkung von Konturen an „Sprungkanten“* (c0-Unstetigkeiten, Intensitätssprünge). In der dunklen Fläche zeigt sich eine dunklere, in den hellen Flächen eine hellere Kantenlinie. Dort, wo Konturen vorhanden sind, ist das *vorteilhaft* (evolutionäre Entwicklung der menschlichen visuellen Wahrnehmung), obwohl Täuschungen damit verbunden sind (photometrischer Eindruck).
- Bei *Sprüngen in der Helligkeitsänderung* (c0-Stetigkeit, c1-Unstetigkeit, typisch für Approximation durch ebene Polygone beim Gouraud-Shading, z.B. Zylinder) *stört dieser Effekt u. U. erheblich*.

Direkte Schattierungsverfahren (Mach-Band-Effekt an Kanten)



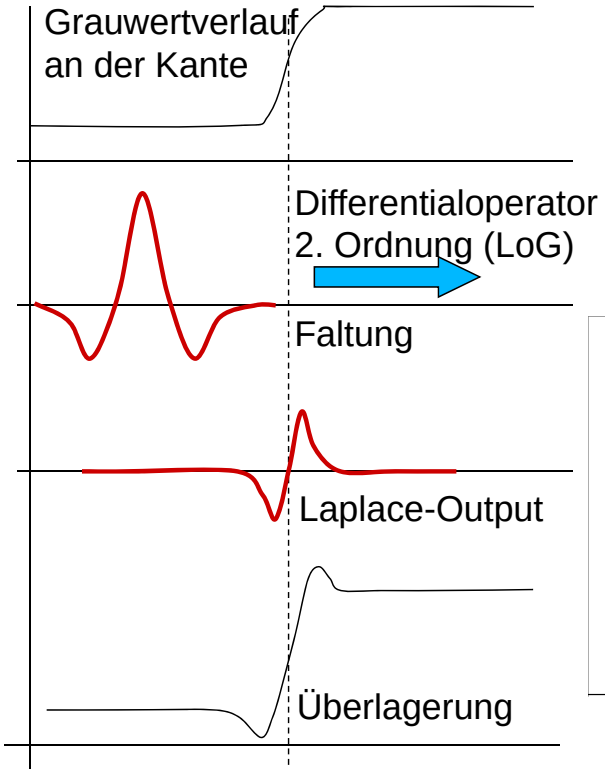


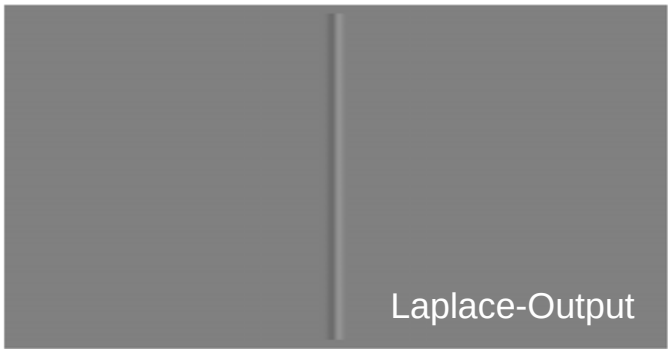
zunächst Kanten: Liegen eine helle und eine dunkle Fläche nebeneinander, beobachtet man einen dunklen Streifen auf der dunkleren Seite und einen hellen Streifen auf der helleren Seite (Kontrastverstärkung).

Direkte Schattierungsverfahren (Mach-Band-Effekt an Kanten)

(Wikipedia): Bei einer Abfolge von Flächen unterschiedlicher Graufärbung, die in sich keine Farbgraduierung haben, beobachten wir entlang der Grenzen **machsche Streifen** (nach Ernst Mach 1865). Dabei handelt es sich um helle und dunkle Streifen, die den Kontrast zwischen den Flächen verstärken.

Direkte Schattierungsverfahren (Mach-Band-Effekt an Kanten)





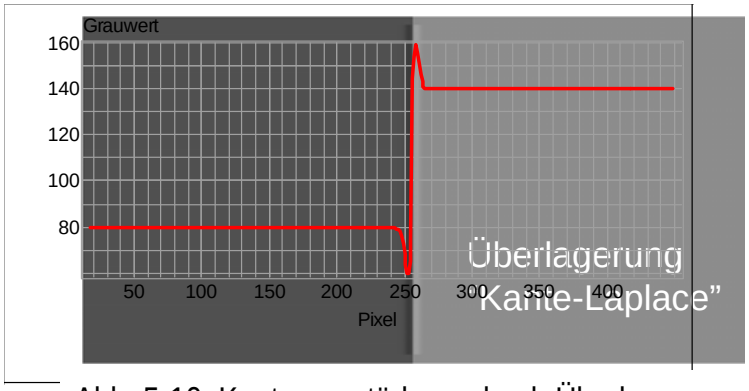
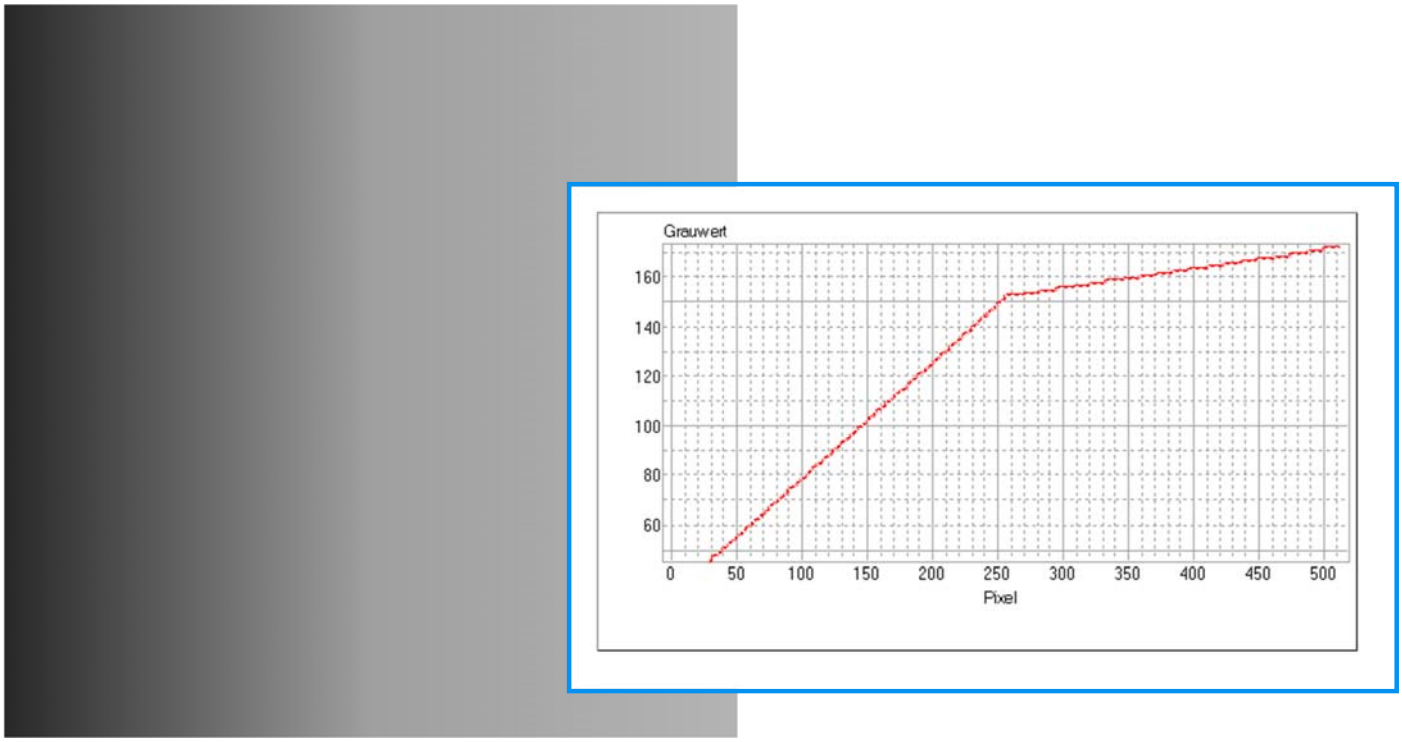


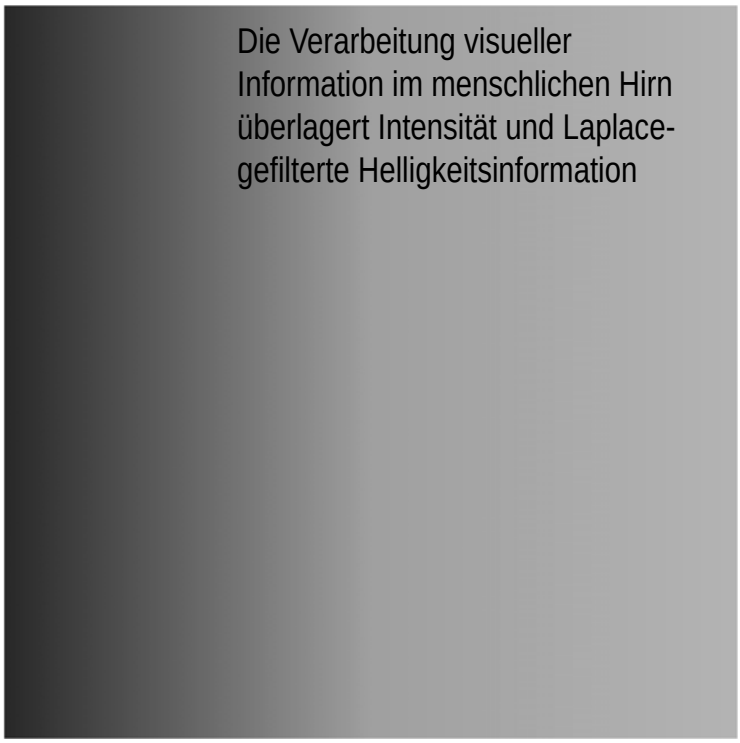
Abb. 5-10: Kantenverstärkung durch Überlagerung von Intensität und zweiter Ableitung (modelliert)

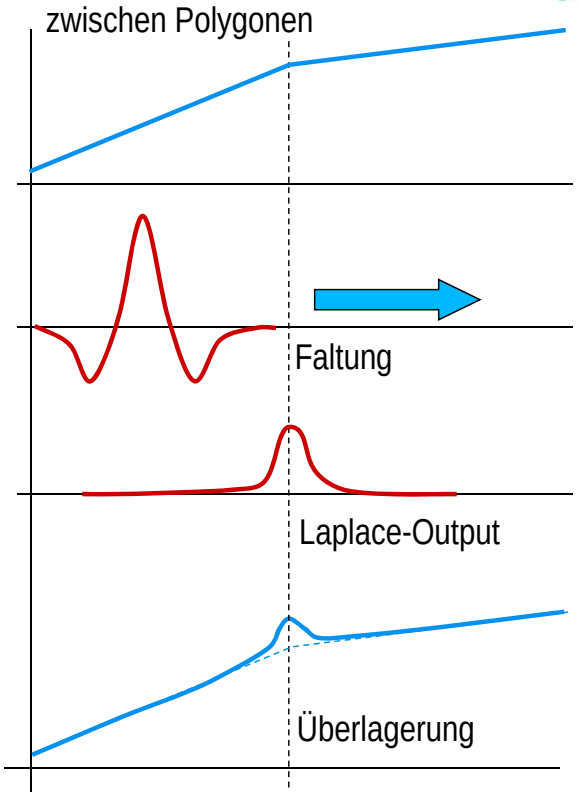
Mach-Band-Effekt beim Gouraud-Shading



Objekt, mit Gouraud-Ansatz schattiert: trotz kontinuierlich ansteigendem Grauwertverlauf sieht man an der Stelle der c1-Unstetigkeit einen helleren Bereich (Bildmitte)

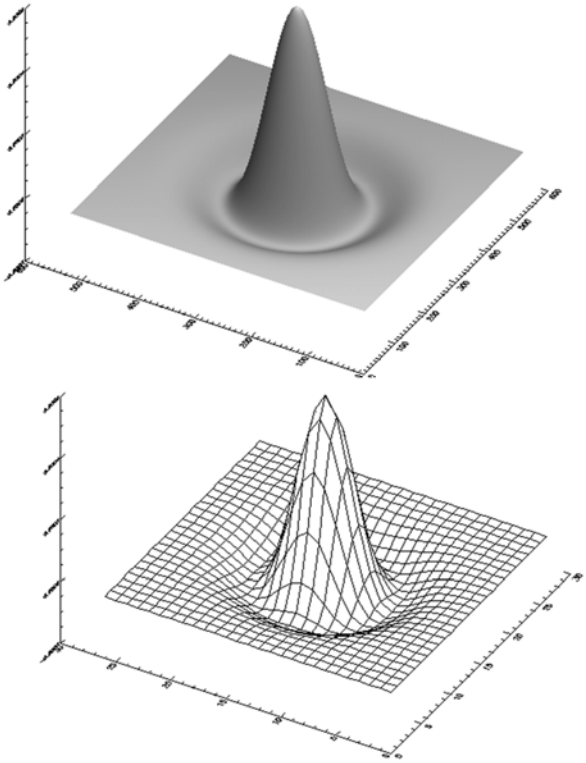
Mach-Band-Effekt beim Gouraud-Shading





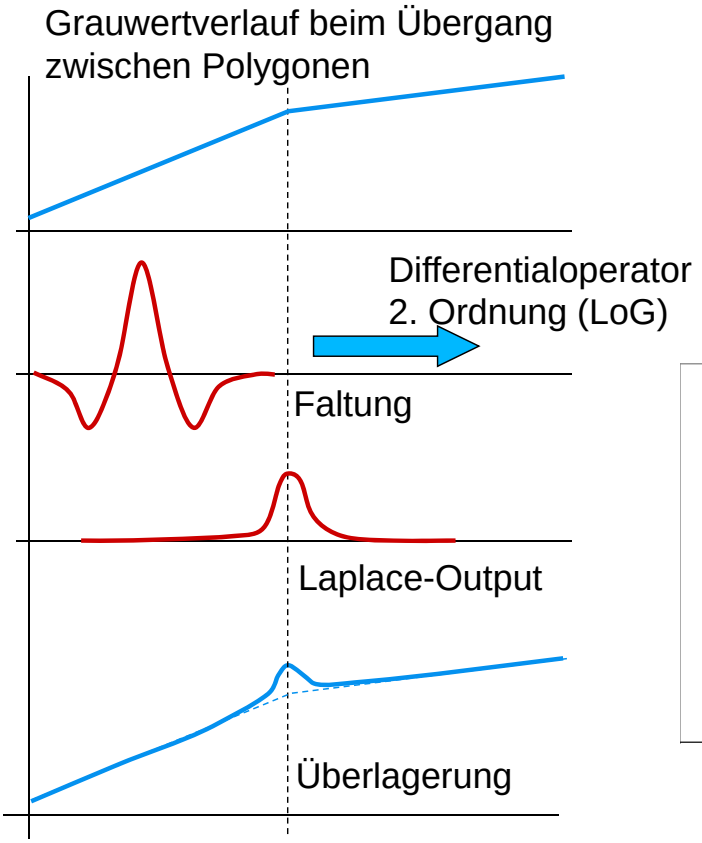
Objekt, mit Gouraud-Ansatz schattiert: trotz kontinuierlich ansteigendem Grauwertverlauf sieht man an der Stelle der c1-Unstetigkeit einen helleren Bereich (Bildmitte)

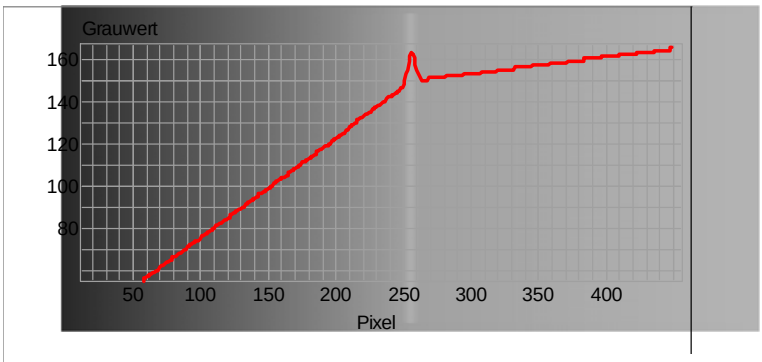
Mach-Band-Effekt beim Gouraud-Shading



Zur Modellierung des Mach-Band-Effektes verwendetes LoG-Filter (σ=3 → win =27, oben schattiert, unten im Abtastraster)

Mach-Band-Effekt beim Gouraud-Shading



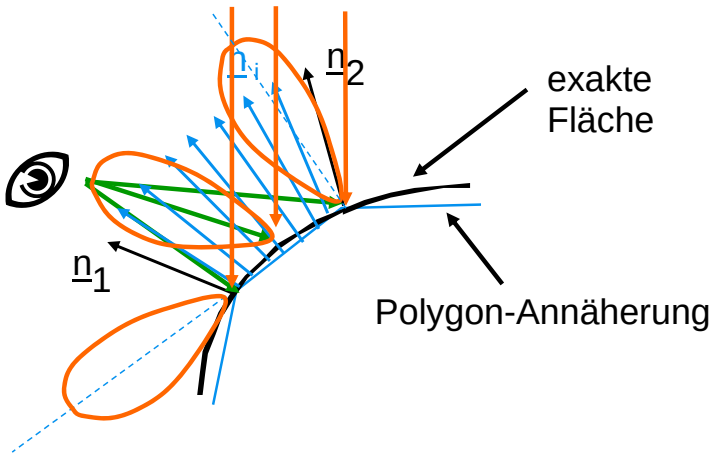


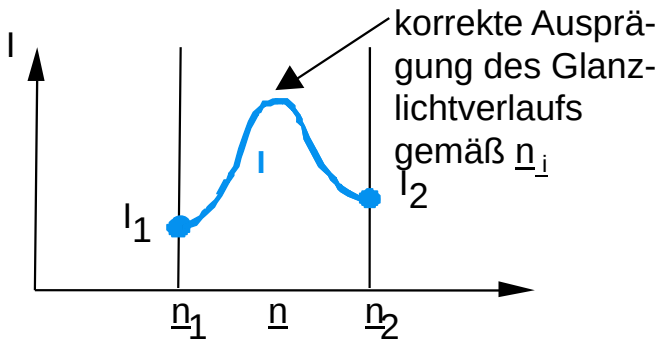
Störung der schattierten Fläche durch Überlagerung von Intensität und zweiter Ableitung (modelliert)

Direkte Schattierungsverfahren (Phong-Shading)

Phong-Shading *[Phong 1975]:*

- Lineare Interpolation der Normalenvektoren zwischen den Polygonecken anstelle von Interpolation der Intensitätswerte (bei Grafikkarten/-software als Pixelshader bekannt).
- Exakte Berechnung der $\cos^n$-Funktion im Phong-Beleuchtungsmodell für jedes Pixel : Glanzlichter werden erhalten, Mach-Band-Effekt wird vermieden.





Interpolation der Normalenvektoren

Direkte Schattierungsverfahren (3D-Grafikhardware)

3D-Grafikhardware

Was wird üblicherweise generell von der Hardware unterstützt?

- Z-Puffer-Verfahren für Flat-Shading bzw. Gouraud-Shading
- Double Buffering
- Performance: mehrere 100.000 Polygone pro Sekunde mit korrekter Sichtbarkeit und Beleuchtung

Pipeline-Architektur der Grafikhardware:

- Die Ausführung vieler Operationen erfolgt nur einmal pro Pixel.
  (Inkrementelle Berechnung des z-Wertes mit Ganzzahlarithmetik, Vergleich der z-Werte mit dem Z-Puffer, Interpolation der Farbwerte beim Gouraud-Shading, Mischen der Farbwerte gemäß der α - Werte, Abspeicherung der neuen Werte)

→ *Echte Parallelität mehrerer Pipelines* und
→ *quasiparallele Ausführung* der o.g. sequentiellen Prozesse *innerhalb der Pipelines* (Parallelität in der Zeit, multible instructions)
ermöglicht *vergleichbare Performance der hardware-basierten 3D-Schattierung mit derjenigen beim Füllen von Polygonen in 2D*.

→ *Bei neueren Grafikkarten* gilt dies auch unter Berücksichtigung von Funktionserweiterungen wie *anisotropes Textur-Mapping und Pixelshading (Phong-Shading).*

Direkte Schattierungsverfahren (3D-Grafikhardware)

Übliche Performanzangaben für Grafik-Hardware:

- *Geschwindigkeitsangaben (Polygonleistung) für normierte Polygone* (z. B. Dreiecke mit 100 Pixeln). Für größere Polygone *sinkt die Performance mit der Anzahl der Pixel* etwa umgekehrt proportional zum Flächeninhalt (da der Hauptaufwand bei der Pixelschattierung liegt).
- Neben der Polygonleistung wird deshalb *meist die Anzahl der abgearbeiteten Pixel pro Zeiteinheit*, (typischerweise mehrere Millionen pro Sekunde) angegeben.

→ *Unterteilung eines großen Polygons in viele kleinere Dreiecke bringt kaum Performance-Einbußen, jedoch Vorteile bei der Berechnung der Beleuchtung (Phong-Beleuchtungs-rechnung für viele Vertices / Glanzlichter / Mach-Band/Effekt).*

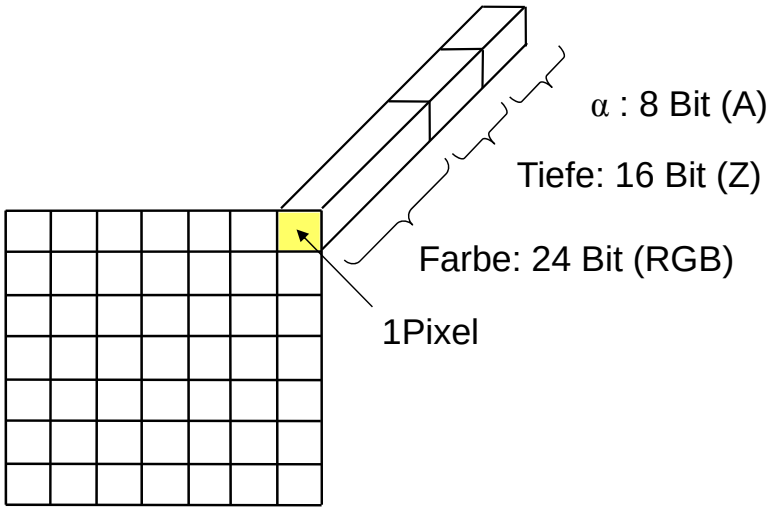
Phong-Shading:

- *Bis 2005* von OpenGL und Hardware aus Effizienzgründen meist *nicht unterstützt*.
- *Seit dem ist aber hardwarebasiertes Phong-Shading verfügbar* (Per-Pixel-Shading), selbst bei einigermaßen kostengünstigen Graphikkarten.

Direkte Schattierungsverfahren (3D-Grafikhardware)

Speicheranforderungen:

- Pro Pixel typischerweise mindestens *48 Bit*: R+G+B+Z+α.
- *1.000.000 Pixel* Bildschirminhalt benötigen *ca. 6 MB* Speicher.



Bildspeicher mit RGB_Z_α-Werten