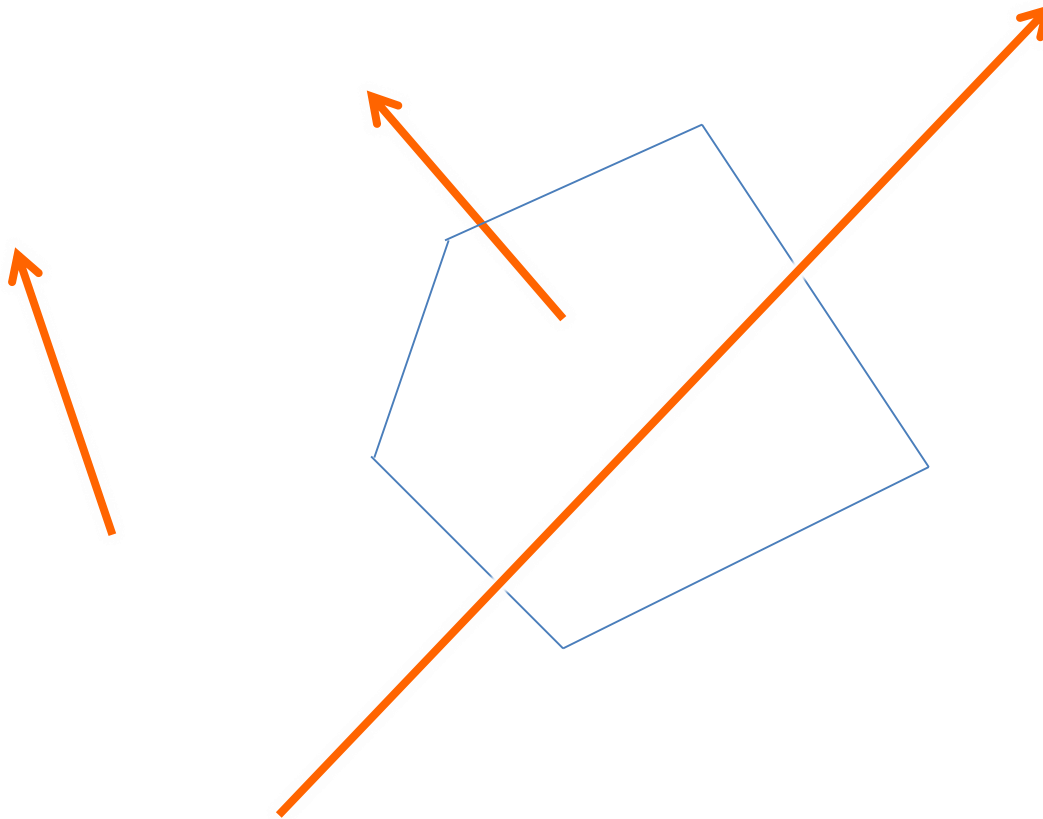


Übung: Interaktive Computergrafik

Wdh: Schattenvolumen



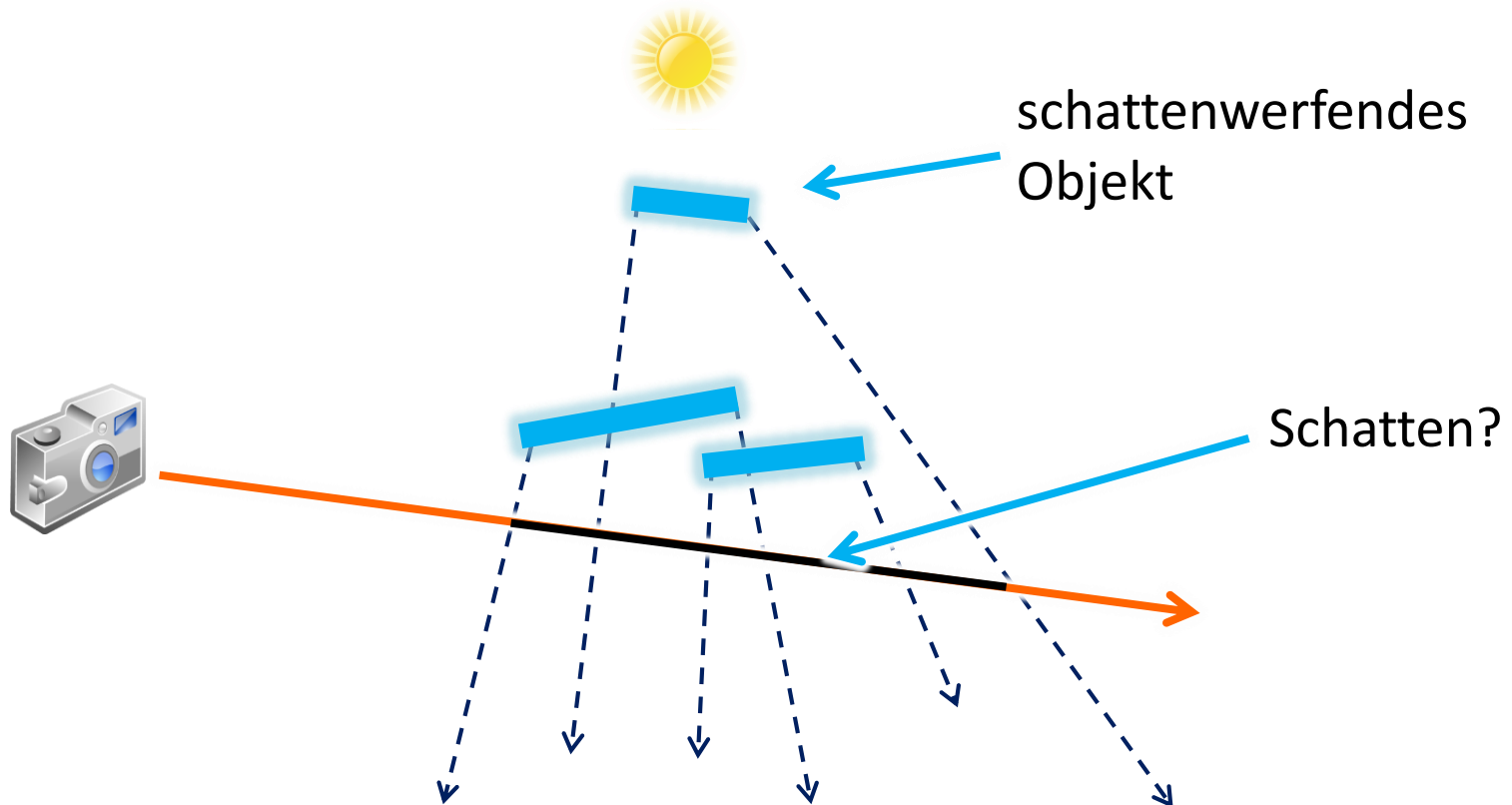
- ▶ Zähle Schnittpunkte um rauszufinden ob Strahlursprung innerhalb eines Polygons liegt



Wdh: Schattenvolumen: Schattentest



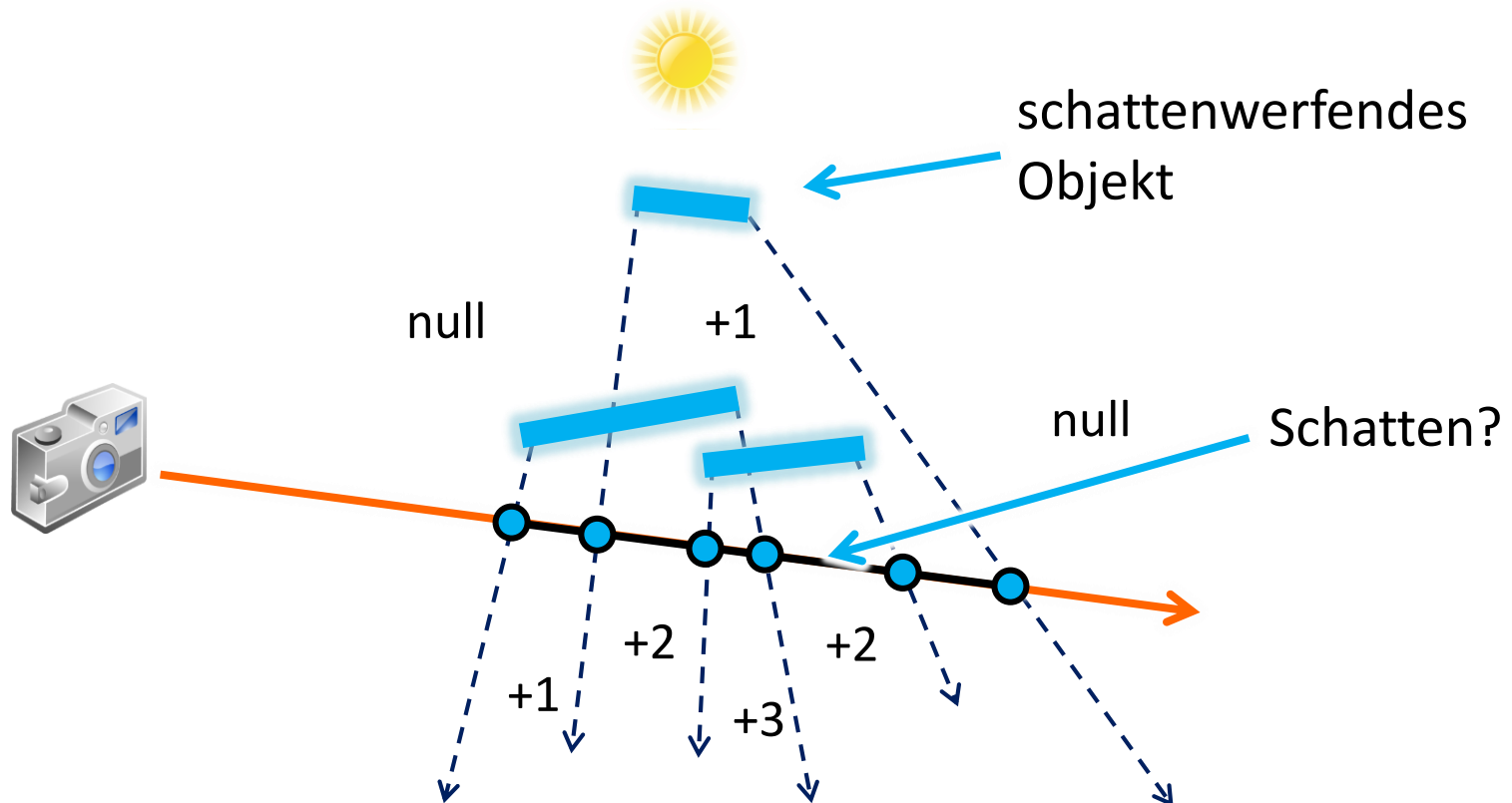
- ▶ verfolge einen Strahl vom Betrachter zum Oberflächenpunkt
- ▶ setze einen Zähler zu Beginn auf Null
- ▶ beim Eintritt in ein SV: inkrementiere Zähler
- ▶ beim Verlassen eines SV: dekrementiere Zähler
- ▶ an der Oberfläche: Punkt befindet sich im Schatten, wenn Zähler > 0



Wdh: Schattenvolumen: Schattentest



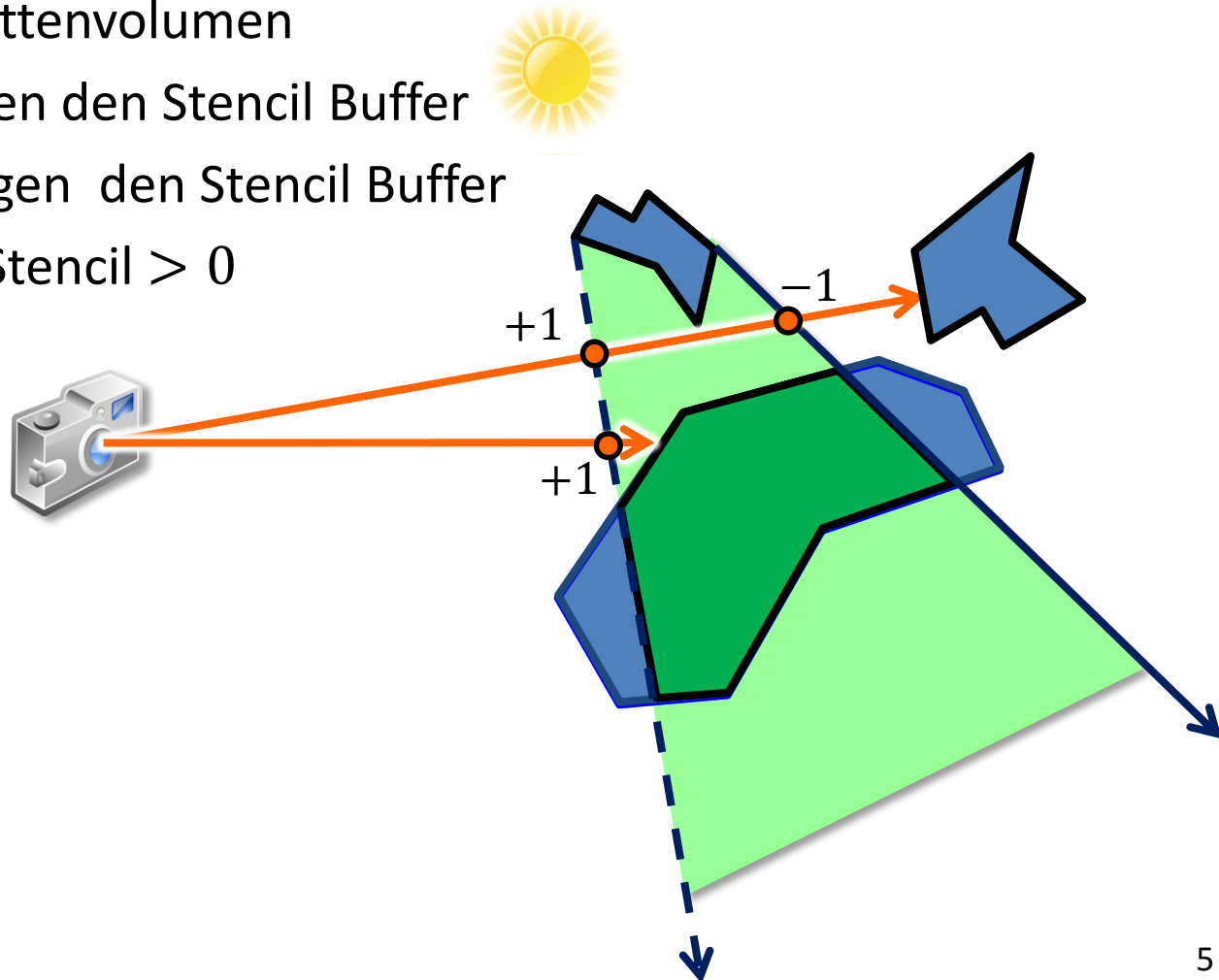
- ▶ verfolge einen Strahl vom Betrachter zum Oberflächenpunkt
- ▶ setze einen Zähler zu Beginn auf Null
- ▶ beim Eintritt in ein SV: inkrementiere Zähler
- ▶ beim Verlassen eines SV: dekrementiere Zähler
- ▶ an der Oberfläche: Punkt befindet sich im Schatten, wenn Zähler > 0



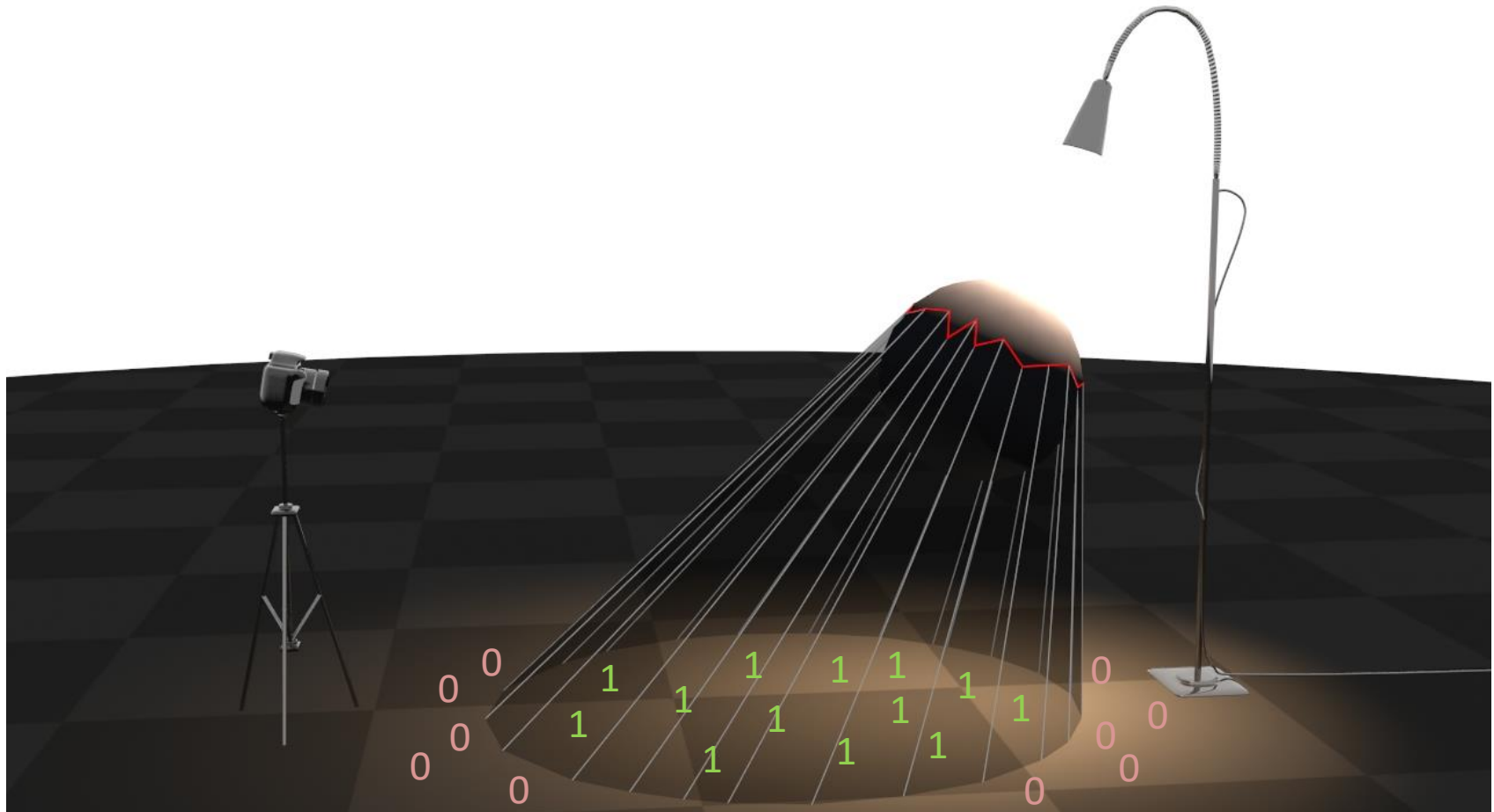
Wdh: Schattenvolumen: Schattentest

Stencil Buffer zum Zählen der Schnitte mit Schattenvolumen

- ▶ zuerst werden die Objekte gezeichnet
- ▶ anschließend wird Schreiben in den Farb- und Tiefenpuffer deaktiviert
- ▶ zeichne dann die Schattenvolumen
 - ▶ Vorderseiten erhöhen den Stencil Buffer
 - ▶ Rückseiten erniedrigen den Stencil Buffer
- ▶ Schatten ist dort, wo $\text{Stencil} > 0$



Wdh: Shadow Volumes



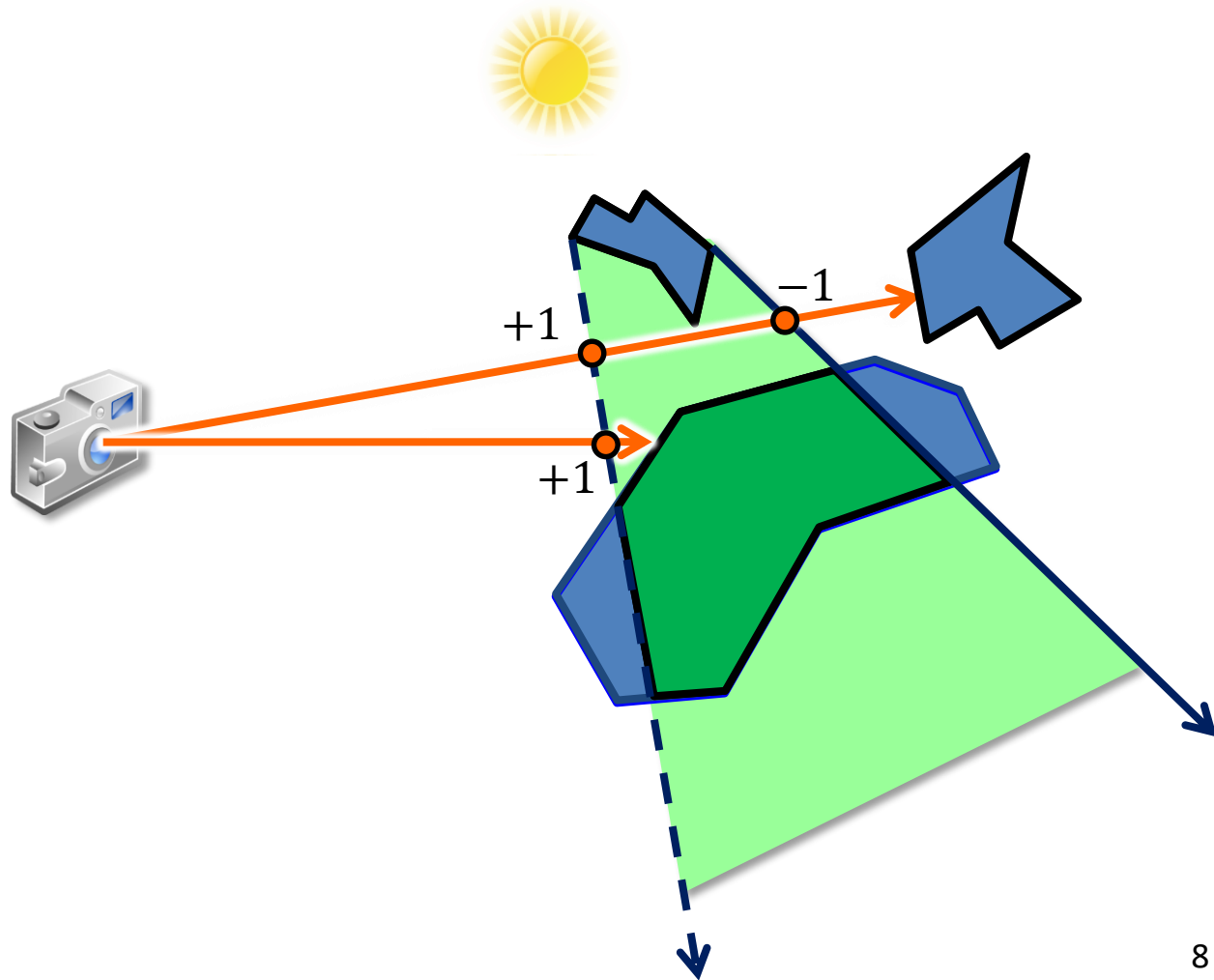
Wdh: Shadow Volumes



Schattenvolumen: Schattentest

Stencil Buffer zum Zählen der Schnitte mit Schattenvolumen

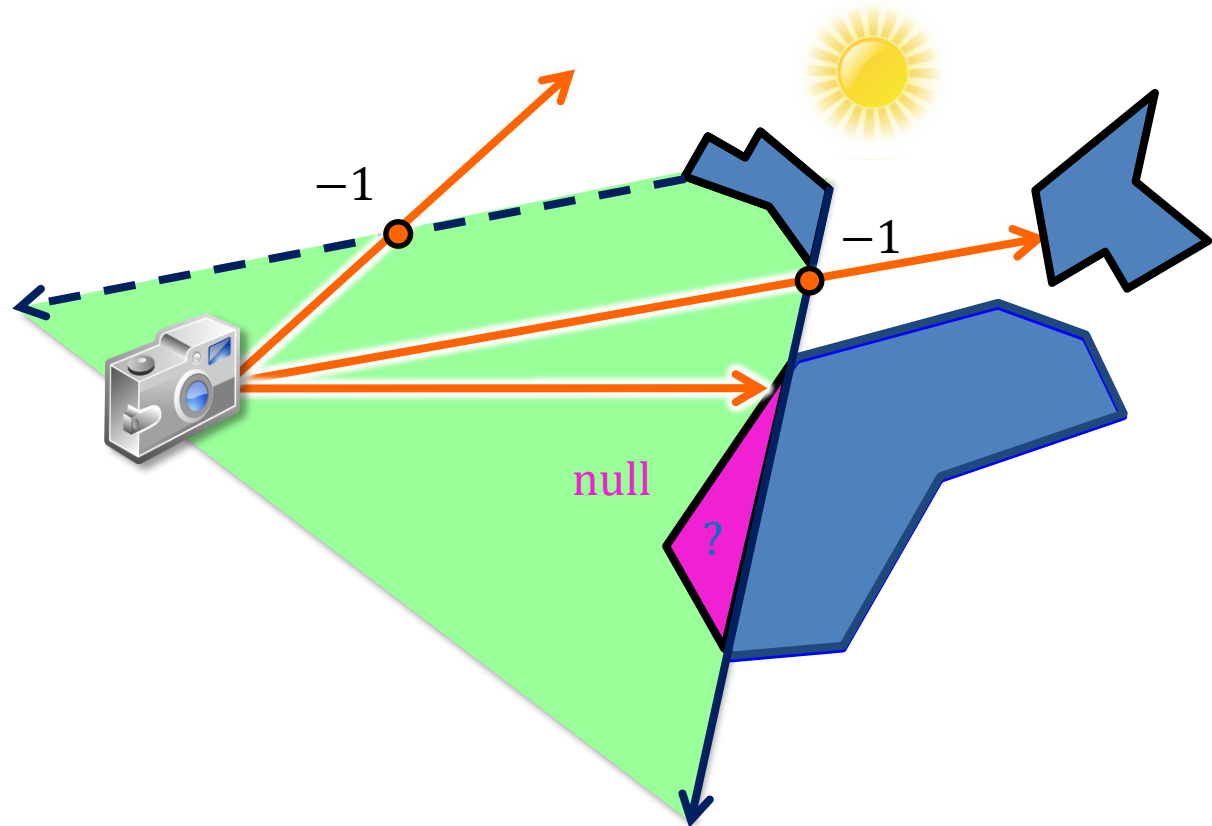
► Funktioniert bisher erstmal nicht für alle Kamera- / Lichtpositionen!



Schattenvolumen: Schattentest

Stencil Buffer zum Zählen der Schnitte mit Schattenvolumen

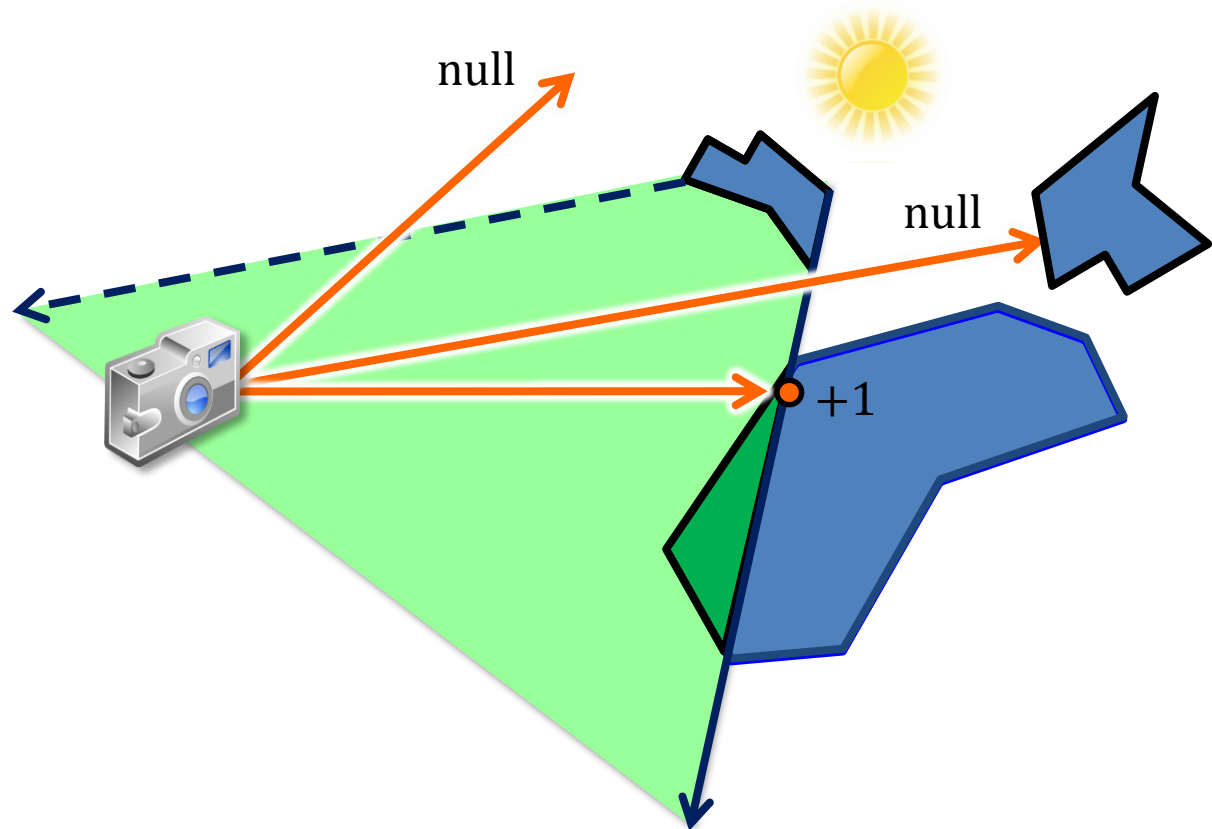
► Funktioniert bisher erstmal nicht für alle Kamera- / Lichtpositionen!



Schattenvolumen: der z-fail Algorithmus

Stencil Buffer zum Zählen der Schnitte mit Schattenvolumen

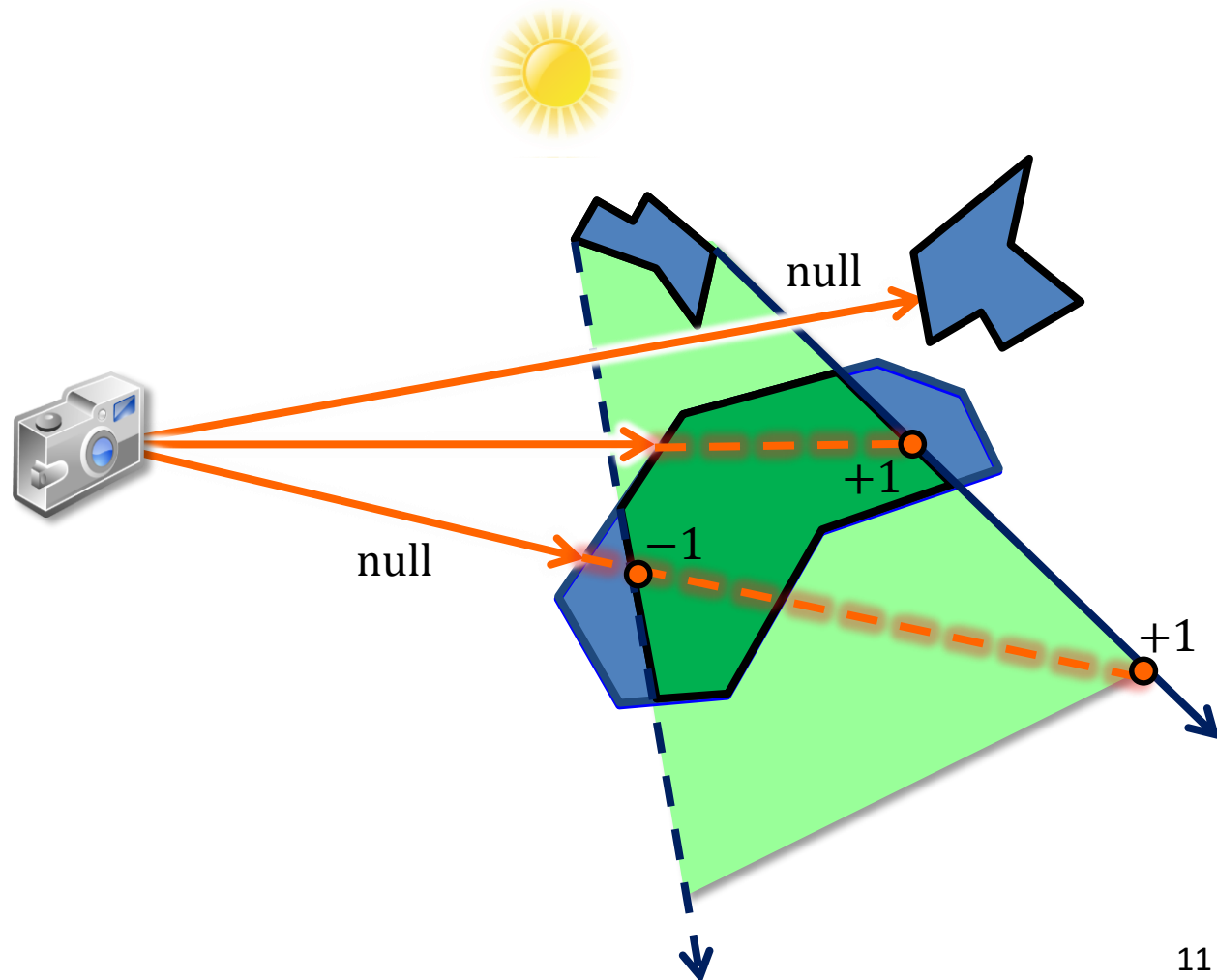
- ▶ Zähle stattdessen **verdeckte** Eintritts- und Austrittspunkte
- ▶ Schatten ist dort, wo $\text{Stencil} > 0$



Schattenvolumen: der z-fail Algorithmus

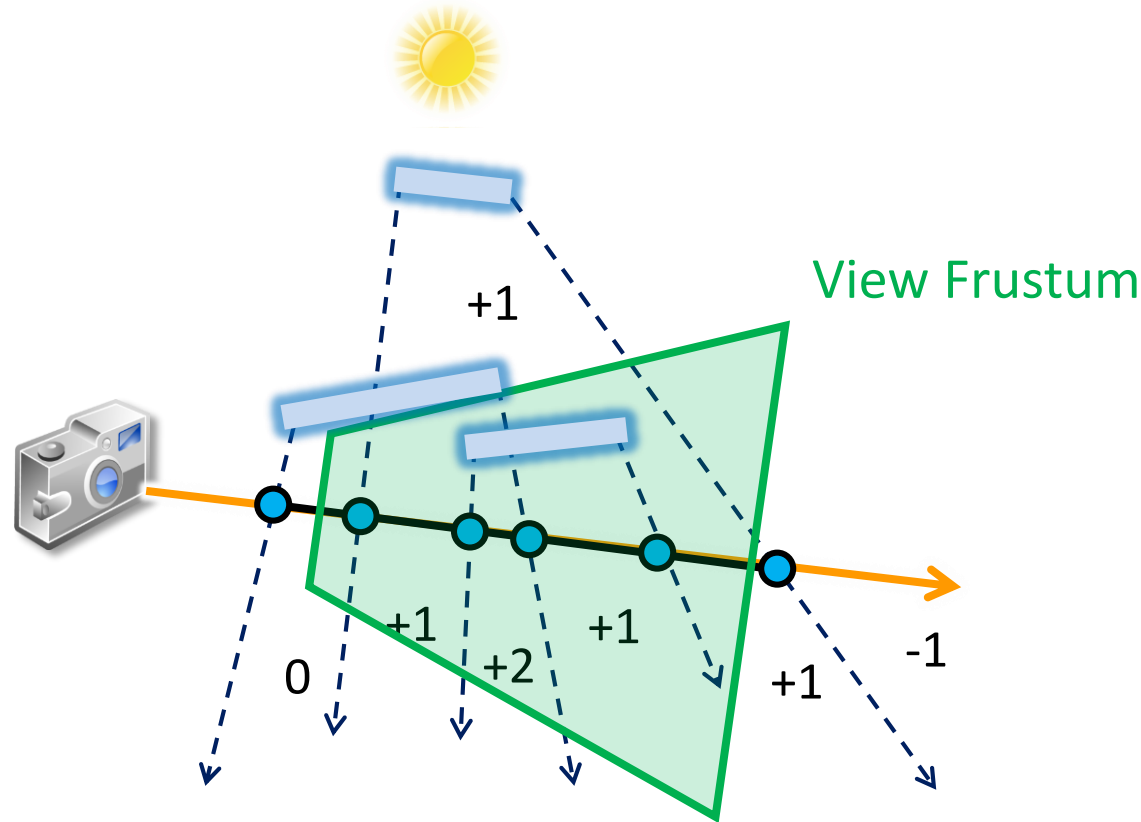
Stencil Buffer zum Zählen der Schnitte mit Schattenvolumen

- ▶ Zähle stattdessen **verdeckte** Eintritts- und Austrittspunkte
- ▶ Schatten ist dort, wo Stencil > 0



Schattenvolumen: der z-fail Algorithmus

- ▶ Jetzt Frustum Clipping problematisch
 - ▶ Volumen müssen abgeschlossen sein
- ▶ Mögliche Lösung: Far Plane im unendlichen
 - ▶ Modifizierte Projektionsmatrix



„Unendliche“ Projektionsmatrix



- was passiert in der Projektionsmatrix, wenn wir die Far Plane ins Unendliche verschieben, also $f \rightarrow \infty$?

$$\begin{pmatrix} 2n(r-l) & 0 & (r+l)/(r-l) & 0 \\ 0 & 2n(t-b) & (t+b)/(t-b) & 0 \\ 0 & 0 & -(f+n)/(f-n) & -2fn/(f-n) \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

► $\frac{f+n}{f-n} \rightarrow 1$ und $\frac{2fn}{f-n} \rightarrow 2n$

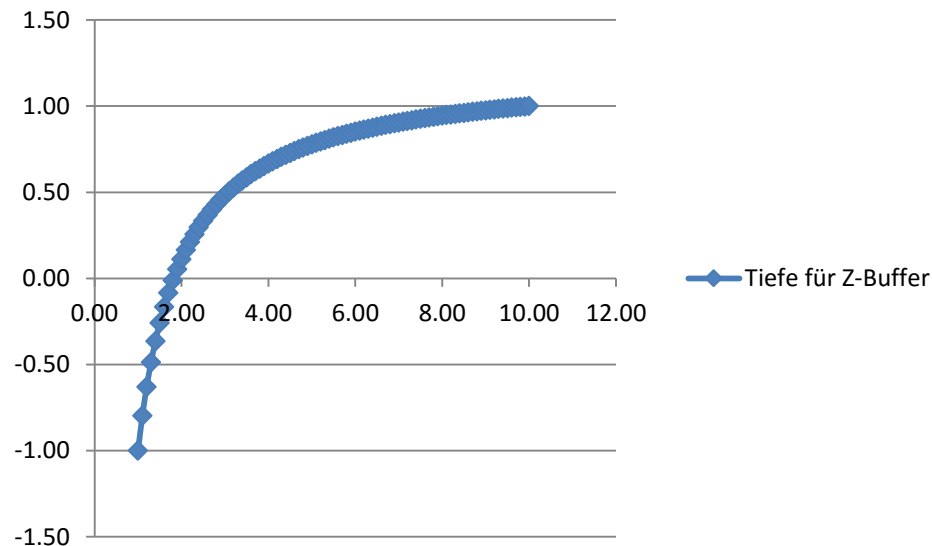
- für die Matrix ergibt sich dementsprechend:

$$\begin{pmatrix} 2n(r-l) & 0 & (r+l)/(r-l) & 0 \\ 0 & 2n(t-b) & (t+b)/(t-b) & 0 \\ 0 & 0 & -1 & -2n \\ 0 & 0 & -1 & 0 \end{pmatrix}$$

„Unendliche“ Projektionsmatrix



- ▶ Far Plane repräsentiert also tatsächlich unendlich weit entfernte Punkte
- ▶ der Genauigkeitsverlust im Tiefenpuffer (für andere entfernte Flächen) ist akzeptabel, da die Abbildung der Tiefe ohnehin nicht linear ist
- ▶ wichtig für den z-fail Algorithmus ist nur: die Flächen werden tatsächlich rasterisiert



Schattenvolumen: der z-fail Algorithmus



- ▶ beim z-fail Algorithmus müssen die Schattenvolumen auch „hinten“ geschlossen sein!
- ▶ hinter dem betrachteten Punkt auf der grünen Fläche würde sonst nie ein Fragment rasterisiert werden, das den Tiefentest nicht besteht
- ▶ deshalb hat das Schattenvolumen eine **Rückseite**

