

FOTOREALISTISCHE VISUALISIERUNGEN

Mit Blender

2



6. – 8. NOVEMBER 2025

LAG SEMINAR GÖTTINGEN

Matthias Demand

INHALT

1.	RENDERING	X
	Biased VS Unbiased Rendering	X
	Colormanagement	X
	Rendersettings	X
	Realtime Compositor	X
	Objektivfehler	X
	Bokeh	X
	Kameras und selektive Schärfe	X
	ID-Masks	X
2.	LICHT	X
	Lichtobjekte	X
	Emission-Shader	X
	Blackbody	X
	IES Maps	X
	Lightmaps	X
	Gobos	X
	Lightlinking	X
	Lightgroups	X
	Sky & Sun	
	Beleuchten mit HDRI	

3.	MODELS	X
	Hardsurface VS Softbody modeling	X
	Grundlegende Modellierungstools	X
	Bevel Modifier	X
	Subdivision Surface Modifier	X
	Sculpting	X
4.	SHADER	X
	Was ist Shading	X
	Normals	X
	PBR Workflow	X
	Node Wrangler	X
	BSDF Shader	X
	Roughness in Blender 4.3	X
	Glassshader	X
	SubSurfaceScattering	X
	UV Unwrapping	X
5.	ASSET-MANAGER	

RENDERING 1

BIASED VS UNBIASED RENDERING

1. Biased Renderer / Eevee

Biased Renderer schummeln. Sie verwenden bestimmte Annahmen und Optimierungen, um das Rendering zu beschleunigen. Sie sind darauf ausgelegt, die Bildqualität effizient zu verbessern, ohne jede Lichtinteraktion genau zu simulieren. Durch das „Biasing“, also die bewusste Verzerrung oder Vereinfachung, wird die Rechenzeit verkürzt. Biased Renderer verwenden Techniken wie Approximationen und Interpolationen, um die Genauigkeit zu verringern und die Renderzeit zu verkürzen.

Approximation bedeutet, dass ein komplexer Wert oder eine Berechnung durch eine näherungsweise Darstellung ersetzt wird. Anstatt eine exakte Lösung für eine Berechnung zu finden, wird eine Schätzung verwendet, die „gut genug“ ist und sich schneller berechnen lässt. In der 3D-Grafik und beim Rendering bedeutet das, dass bestimmte Aspekte der Lichtsimulation oder Oberflächenberechnung vereinfacht werden, um die Renderzeit zu verkürzen und Rechenressourcen zu sparen.

2. Unbiased Renderer / CYCLES

Unbiased Renderer verfolgen einen realistischeren Ansatz, indem sie versuchen, Lichtstrahlen so genau wie möglich zu simulieren. Jeder Lichtstrahl wird berechnet, ohne dabei Vereinfachungen oder „Bias“ zu verwenden. Dies führt zu extrem genauen Ergebnissen, allerdings zu höheren Rechenkosten.

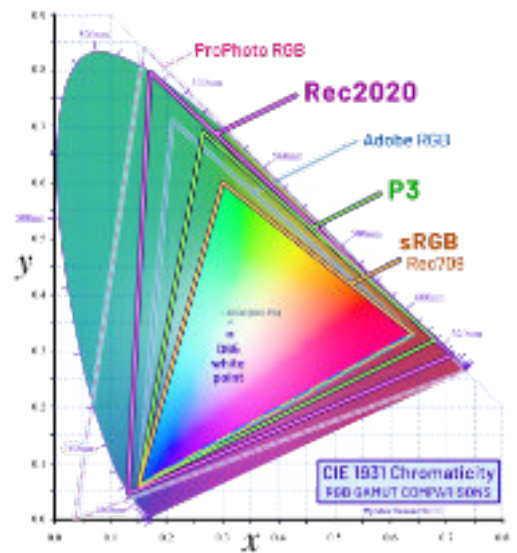
Die Wahl zwischen biased und unbiased hängt also von den Prioritäten des jeweiligen Projekts ab: Wenn Geschwindigkeit und Effizienz wichtig sind, wählst du einen Biased Renderer (EVEE), wenn Realismus im Vordergrund steht, einen Unbiased Renderer (CYCLES).

COLORMANAGEMENT

Für fotorealistische Renderings wird zum einen ein hoher Dynamikumfang, wie auch ein möglichst großer Farbraum benötigt.

Als Standard ist in den Rendersettings sRGB eingestellt. Sowohl den größeren Farbraum wie auch den höheren Dynamikumfang bietet AgX. Um die besten Ergebnisse zu erzielen, sollte in den Rendersettings von Cycles unter Display Device und unter Sequencer Rec.2020 eingestellt werden. Die neutralsten Ergebnisse bietet unter Look, die Einstellung None.

Ein Rendering mit AgX und Rec.2020 liefert gegenüber einem Rendering in sRGB die weitaus besseren Bildergebnisse.



Whitebalance

Im Colormangement bietet Blender seit der Version 4.3 einen komfortablen Weißabgleich mit Pipette. Anzuwenden ist dieser wie in den bekannten Bildbearbeitungsprogrammen. Da sich das Rendering nicht wie ein RAW verhält, bei dem der Weißabgleich in der Nachbearbeitung vorgenommen werden kann, ist darauf zu achten, den Weißabgleich möglichst präzise in den Rrendersettings vorzunehmen. Eine Veränderung des Weißabgleiches am gerenderten Bild führt immer zu einem Verlust von Tonwerten.

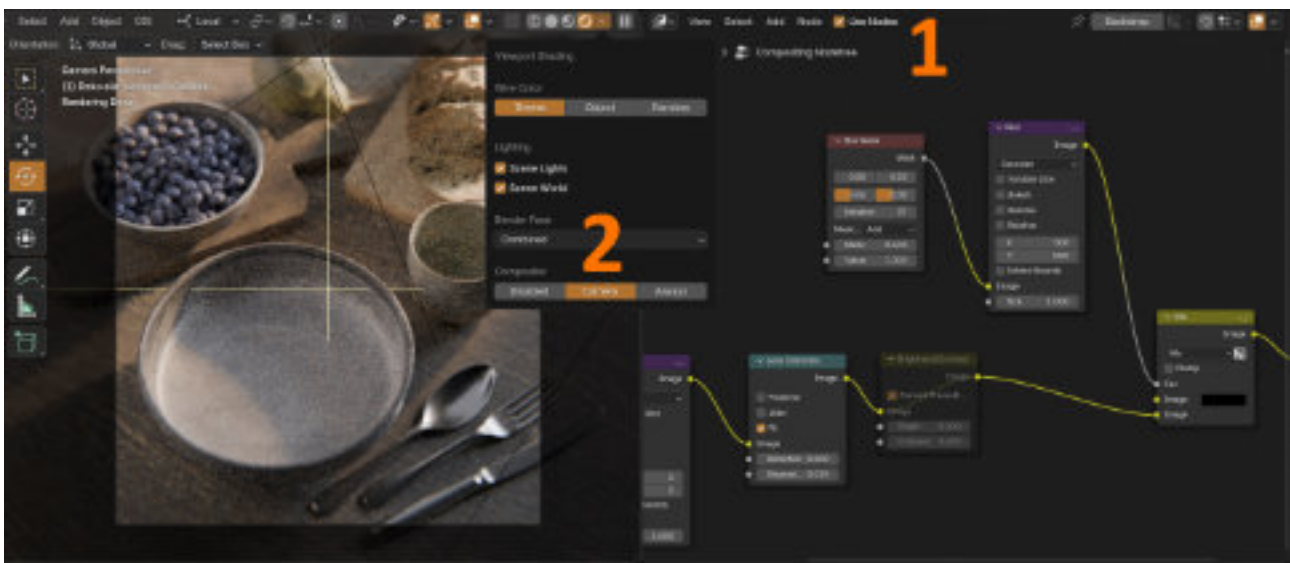
Exposure

Im Bereich Colormangement gibt es einen Exposure Regler, über den die Helligkeit des Bildes korrigiert werden kann. Dieser ist allerdings ein Postproduktion Effekt und hellt das gerenderte Bild auf.

Unter Film gibt es den gleichen Regler noch einmal. Durch eine Korrektur an dieser Stelle entsteht tatsächlich ein helleres bzw. dunkleres Rendering. Eine Belichtungskorrektur ist also in jedem Fall über den exposure Regler unter Film vorzunehmen.

REALTIME COMPOSITOR

Im Compositor lassen sich nicht nur Postproduktion Effekte auf das Rendering anwenden. Durch den Realtime Compositor sind sie direkt im Render-Preview sichtbar. In der Rendervorschau wird somit das fertige Rendering schon mit allen Bearbeitungen und Effekten angezeigt.



Der Compositor ist ein Node basiertes System und hat seine eigene Arbeitsumgebung. Um mit den zugehörigen Nodes arbeiten zu können muß in der oberen Menüzeile des Compositors „use Nodes“ angeklickt sein. (1)

Um die Wirkung der Nodes in Echtzeit zu sehen, muß dies in den Viewport-Shading Einstellungen des Render previews aktiviert werden.

Sinnvoll ist es die Echtzeitdarstellung auf die Kamera zu beschränken. (2) Dann sind alle Compositing Effekte sofort sichtbar, sobald die Kamera aktiviert ist.



OBJEKTIVFEHLER

Zwischen das „render Layer Node“ und das „composite node“ kommen die Nodes für die Post-Produktion-Effekte. Zu diesen zähle ich auch Kontrast- und Farbkorrekturen, also auch den kompletten Bereich Colorgrading.

Helligkeitskorrekturen sollten immer unter Film in den rendersettings vorgenommen werden, da sie als Post-Effekt am fertigen Rendering immer mit einer Verringerung der Tonwerte einhergehen und daher vor dem Rendering stattfinden sollten.

OBJEKTIVUNSCHÄRFTE: Das „Blur Node“ um das Bild ein klein wenig weichzuzeichnen, denn kein Objektiv ist hundertprozentig scharf.

CHROMATISCHE ABERATION: Mit dem „Lens Distortion Node“ lässt sich eine leichte chromatische Aberration an den Bildrändern zufügen. Der Wert unter „Dispersion“ sollte irgendwo um 0.01 liegen. Etwas darüber oder darunter wirkt natürlich. Werte darüber sind im Normalfall zu stark.

Um die Wirkung der einzelnen Nodes einschätzen zu können, lassen sie sich mit der Taste M muten.

VIGNETTE :Das Hinzufügen einer Vignette ist ein wenig aufwändiger.

Hierfür wird zunächst ein „Box Mask Node“ benötigt. Die Größe der Maske lässt sich aber die beiden Werte in der Zweiten Zeile einstellen.

Hinter dem „Box Mask Node“ Wird ein „Blur Node“ benötigt, um mit einem Gausschen Weichzeichner die Maske weichzuzeichnen.

Da das „Compositing Node“ nur einen Input hat, braucht es noch einen „Mix Node“. Hier kommt der Baum des Render Layers in den unteren Image-Input und der Baum der Maske steuert den Factor.

Im Farbwähler des oberen Image Inputs muß noch schwarz ausgewählt werden, da die Maske ja abdunkeln soll. Wenn weiß gewählt ist, wird das Rendering an den Ecken aufgehellt.

Die Intensität der Vignette lässt sich im Box-Mask-Node über den Regler Mask einstellen. Bei einem Wert von 0 hat die vignette ihre maximale Intensität. Je stärker der Wert gegen 1 geht, um so geringer ist die Abdunklung der Ecken.

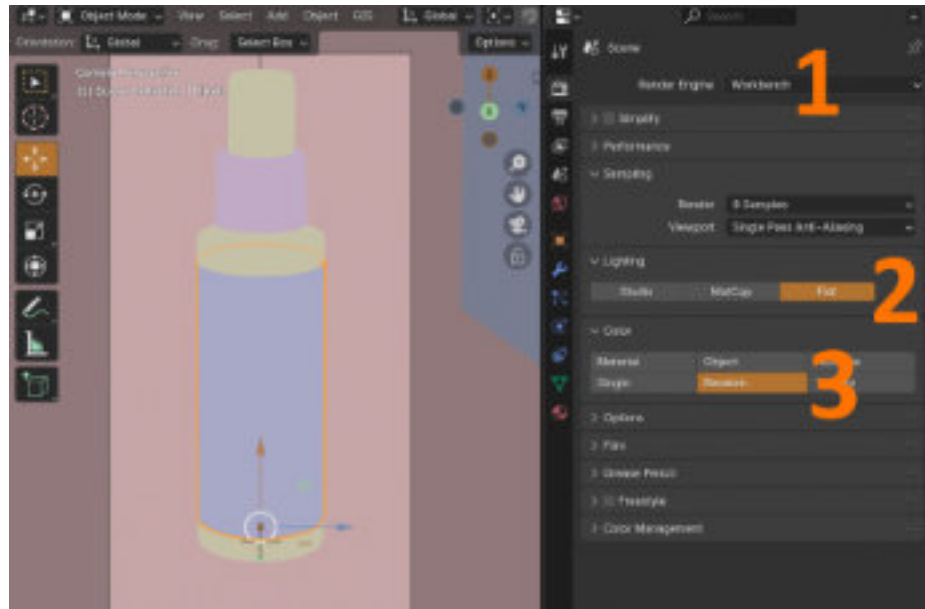
Blendenflecke und Streulicht: Auch diese beiden Effekte gehören in den Bereich der Objektivefehler. Blender stellt mit dem Glare-Node ein Werkzeug zur Verfügung mit dem sich beide Abbildungsfehler realisieren lassen. Die Auswahl „Bloom“ ist zuständig für Streulicht und „Streaks“ produziert Blendensterne.



ID-MASKS

Ein ID-Pass (auch Object oder Material ID-Pass) ist ein separates Rendering, das in der Nachbearbeitung genutzt wird. Es weist Objekten oder Materialien in der Szene eindeutige Farben oder IDs zu, sodass bestimmte Bereiche des Bildes gezielt ausgewählt und bearbeitet werden können.

Um eine ID-Mask zu erstellen muß die Renderengine auf Workbench umgestellt werden. Als nächsten Schritt muß das Lighting auf Flat gestellt und zum Schluß Color auf Random gesetzt werden. Nun die Maske wie ein gewöhnliches Rendering rendern und unter einem eindeutigen Namen abspeichern. Die ID-Maske und das eigentliche Rendering passen pixelgenau aufeinander und können für die Nachbearbeitung in Photoshop in zwei unterschiedliche Ebenen kopiert werden. Mit einem Klick der Pipette lassen sich in der ID-Maske die einzelnen Bereiche des Motivs pixelgenau auswählen und abmaskieren.



Workbench:

Die Workbench-Render-Engine in Blender ist eine spezielle Engine, die hauptsächlich für das Arbeiten in der 3D-Ansicht entwickelt wurde und weniger für das finale Rendern gedacht ist. Sie ist ideal für Szenen, die schnell und ohne hohe Rechenleistung dargestellt werden sollen. Workbench ist optimal für Renderings, die schnelles Feedback brauchen, und für Situationen, wo Details, Geometrie und Form wichtiger sind als Licht und fotorealistische Effekte.



LICHT

2

BELEUCHTEN

DAS STUDIO KOMPLETT ABDUNKELN

In jeder Szene gibt es einen Background. Hierbei handelt es sich um eine Sphäre die sich wie eine Kugel um die komplette Szene spannt. In der Startszene hat dieser Background eine mittleres Grau, was den Effekt hat dass die Szene auch ohne Lichtquellen eine diffuse und dämmerige Beleuchtung hat, da die komplette Sphäre durch die graue Farbe ein wenig Licht abgibt. Hierdurch gibt es eine Grundhelligkeit und dadurch auch eine Aufhellung der Schatten.

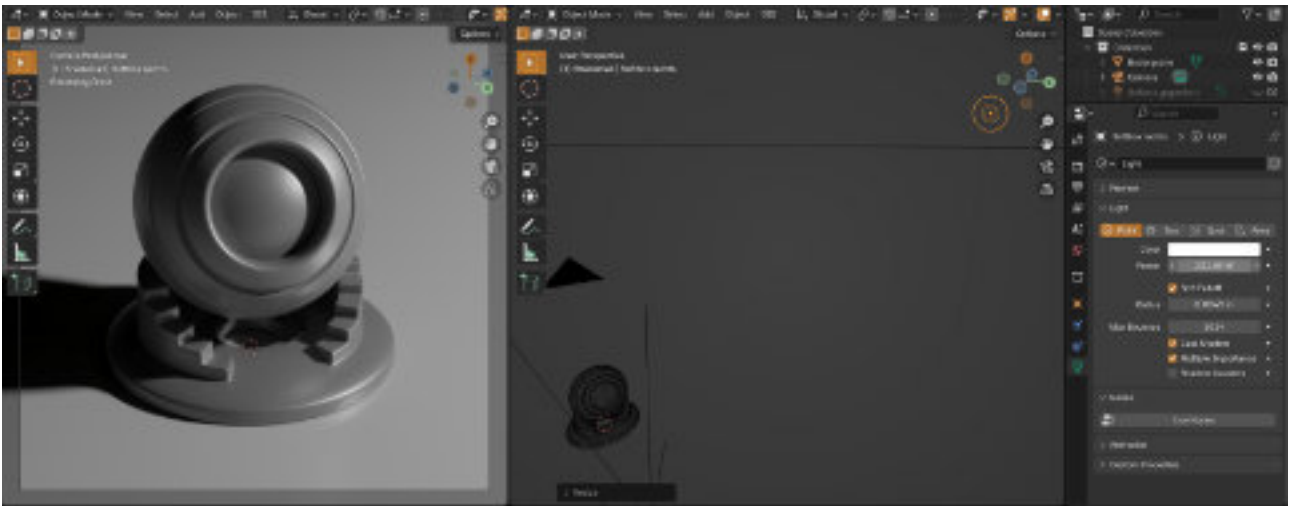
Um präzise Licht zu setzen, ist ein komplett abgedunkeltes Studio notwendig. Hierzu muß in den Properties der Background im Reiter „World“, auf komplett schwarz gesetzt werden.

Wenn keine Lichtquellen vorhanden sind bleiben nun der Renderpreview, wie auch das finale Rendering komplett schwarz.

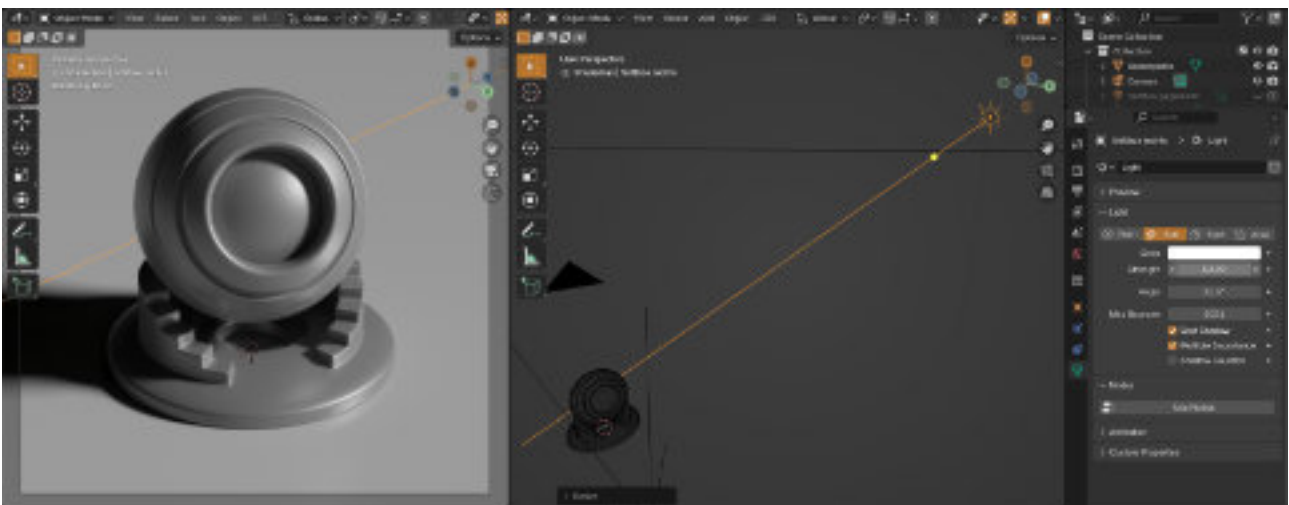
LICHT EMITTIERENDE GRUNDOBJEKTE

Die einfachste Art um eine Szene zu beleuchten sind Lichtquellen, die Blender fix und fertig zur Verfügung stellt. Diese sind in der oberen Menüleiste, bei „Add“ unter „Lights“ zu finden.

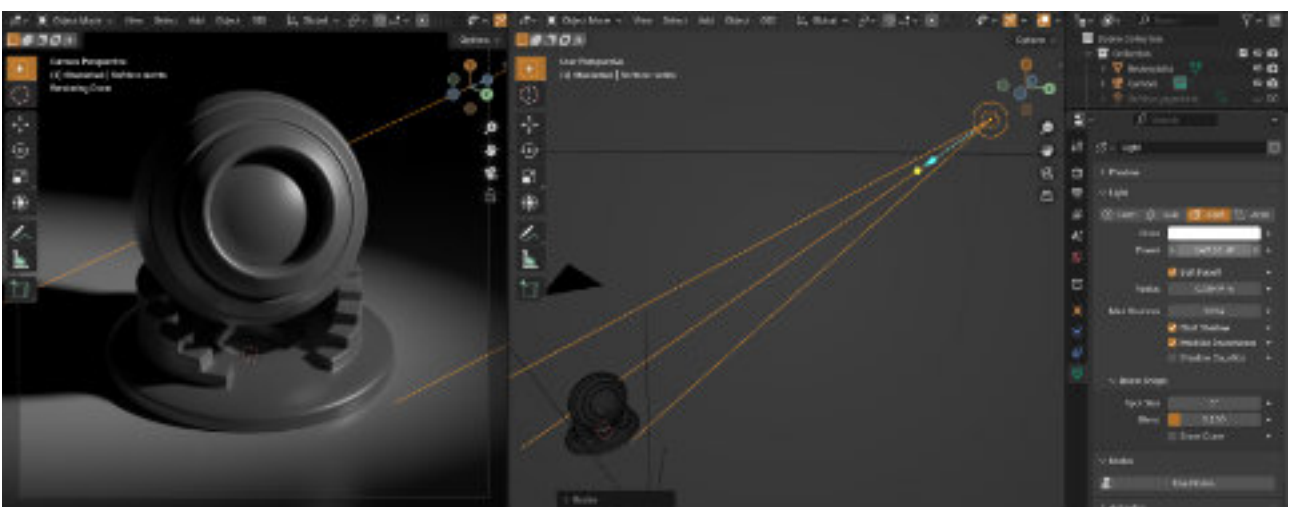
Blender verfügt hier über vier unterschiedliche Lichtquellen:



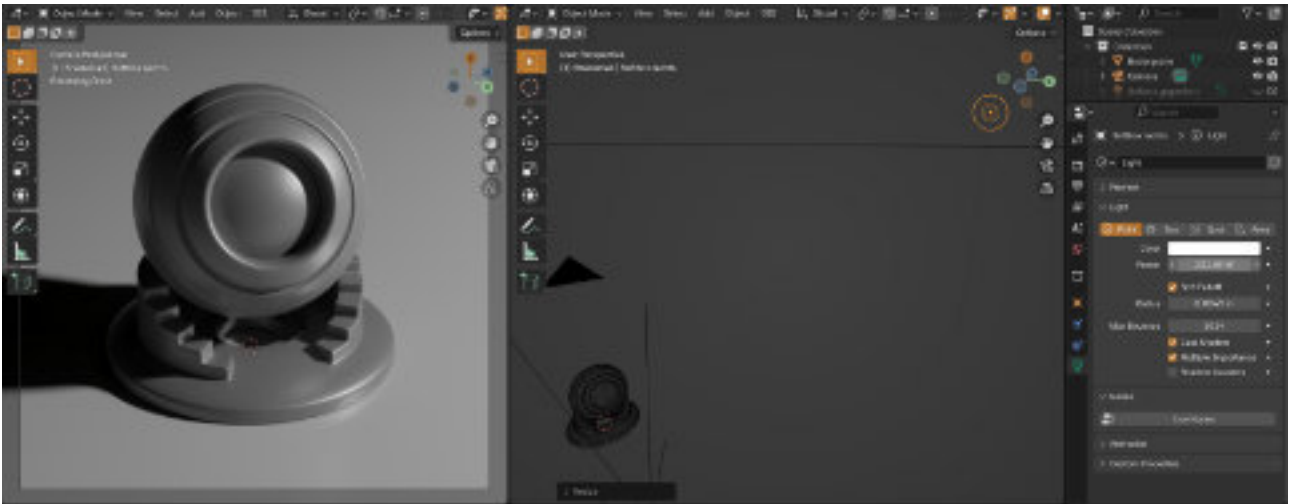
Point: Eine Kugelförmige Lichtquelle, welche Licht in alle Richtungen abgibt.



Sun: Eine Lichtquelle deren Abstrahlverhalten identisch mit dem Point-Light ist. Der Unterschied zwischen den beiden Lichtquellen liegt darin, dass die Beleuchtungsintensität des Point-Light in Abhängigkeit zur Entfernung zu den Objekten einer Szene zu- oder abnimmt. Die Beleuchtungsintensität beim Sun-Light bleibt unabhängig vom Abstand immer gleich.



Spot: Beim Spot handelt es sich um eine gerichtete Lichtquelle, die über einen Lichtkegel verfügt, dessen Größe und Leuchtwinkel sich einstellen lässt. Die Härte der Schatten lässt sich über die Größe des Leuchtmittels regeln.



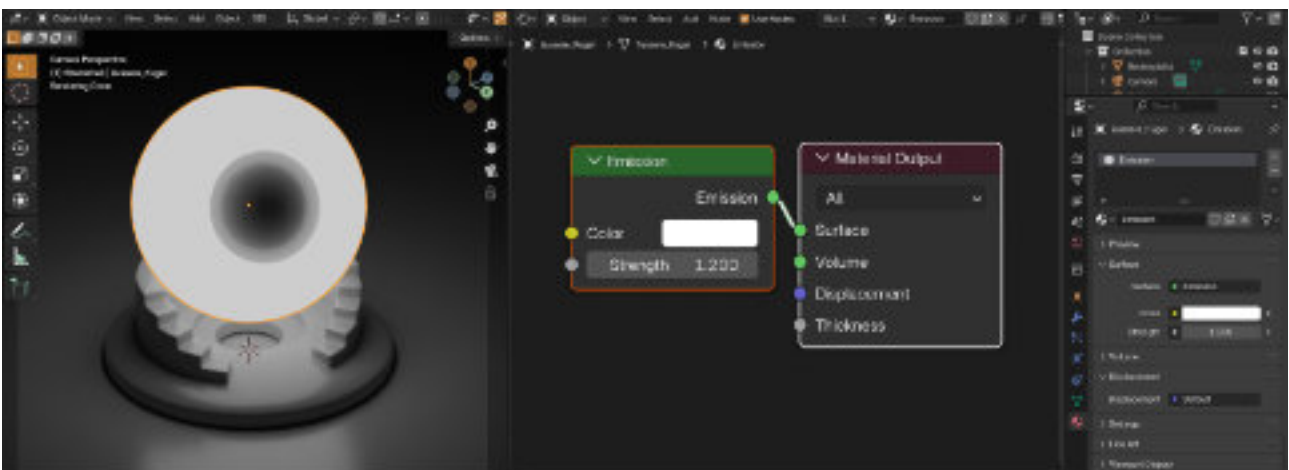
Area: Das Arealight ist eine Flächenleuchte, welche bei zunehmender Größe immer diffuseres Licht produziert und bei abnehmender Größe immer mehr zu punktförmigen Lichtquelle wird, die harte Schatten produziert.

Gemeinsam haben diese Lichtquellen, dass im Rendering nur ihre Lichtwirkung zu sehen ist. Ebenso sind sie als Spiegelung in einer reflektierenden Oberfläche zu sehen. Im Rendering selbst, und auch im Renderpreview sind sie unsichtbar.

LICHTEMITIERENDE OBJEKTE

Werden Objekte mit einem Emission-Shader belegt, fangen sie an zu leuchten. Hierfür muß in der Textur nur der BSDF-Shader durch einen Emission-Shader ersetzt werden.

Diese Lichtquellen, haben nicht nur eine Lichtwirkung, die eine Auswirkung auf das Rendering hat. Sie sind auch im Rendering sichtbar... vorausgesetzt sie sind nicht außerhalb des Bildes.



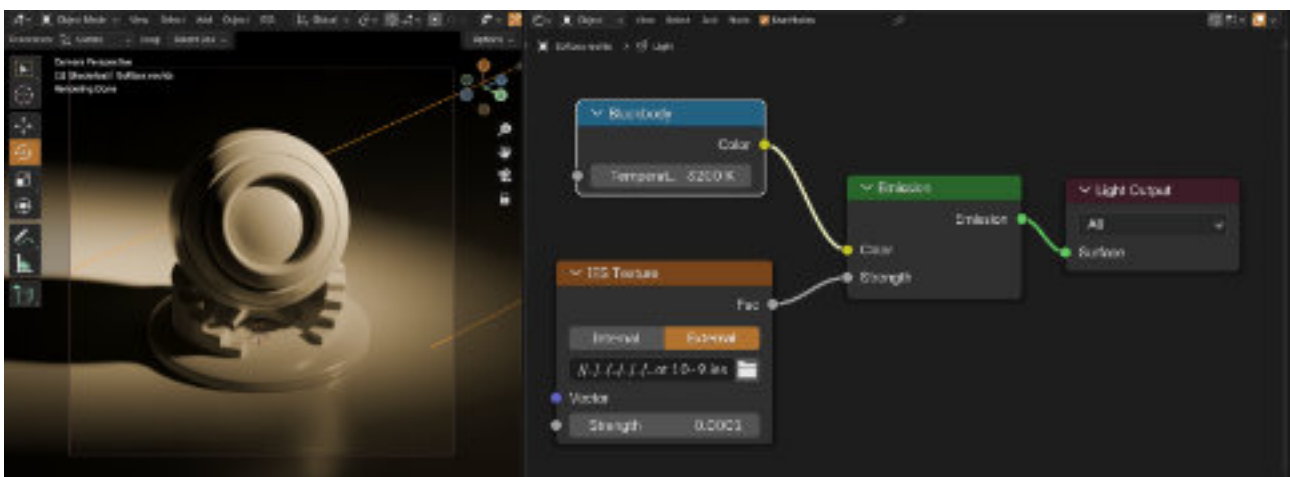
Da beide Typen der oben vorgestellten Lichtquellen mit dem Node-System kompatibel sind, lassen sie sich durch die Verknüpfung mit unterschiedlichen Nodes für eine präzise Lichtführung verwenden.

BLACKBODY NODE

Um die Farbe des abgegebenen Lichts zu verändern, lässt im Emission-Shader unter Color jede Farbe aus dem Farbkreis auswählen. Eine exakte Steuerung der Farbtemperatur des abgegebenen Lichts wird durch die Verwendung eines Blackbody-Nodes erreicht. Damit lässt sich die Farbtemperatur präzise in Grad Kelvin regeln. Der Color-Output des Blackbodys steuert die Farbe des Lichts und wird daher an den Color-Input des Emission-Shaders angeschlossen.

IES-MAPS

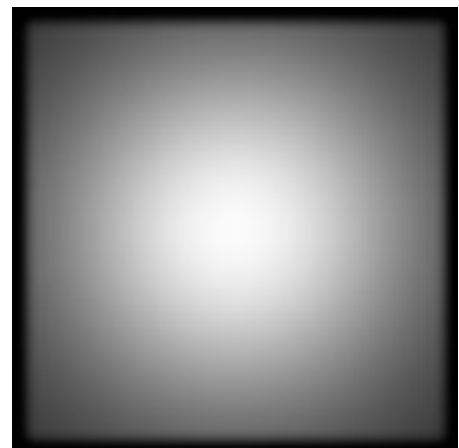
Jede Lichtquelle und jedes Leuchtmittel hat eine ganz individuellen Abstrahlverhalten, bestehend aus Leuchtwinkel, dem Abfall der Lichtintensität von der Mitte bis zum Rand des Lichtkegels und anderen Faktoren. Diese Informationen können in IES Files zusammengefasst werden. Die sicherlich größte Sammlung von IES-Maps ist die Website www.ieslibrary.com. Dort stehen knapp 838.000 IES-Daten von unterschiedlichen Lichtquellen und Leuchtmitteln kostenlos zum Download bereit. Viele Lampenhersteller stellen diese Daten auf ihren Webseiten ebenfalls zur Verfügung. Unter anderem auch Arri für seine legendären Fresnell-Spots.

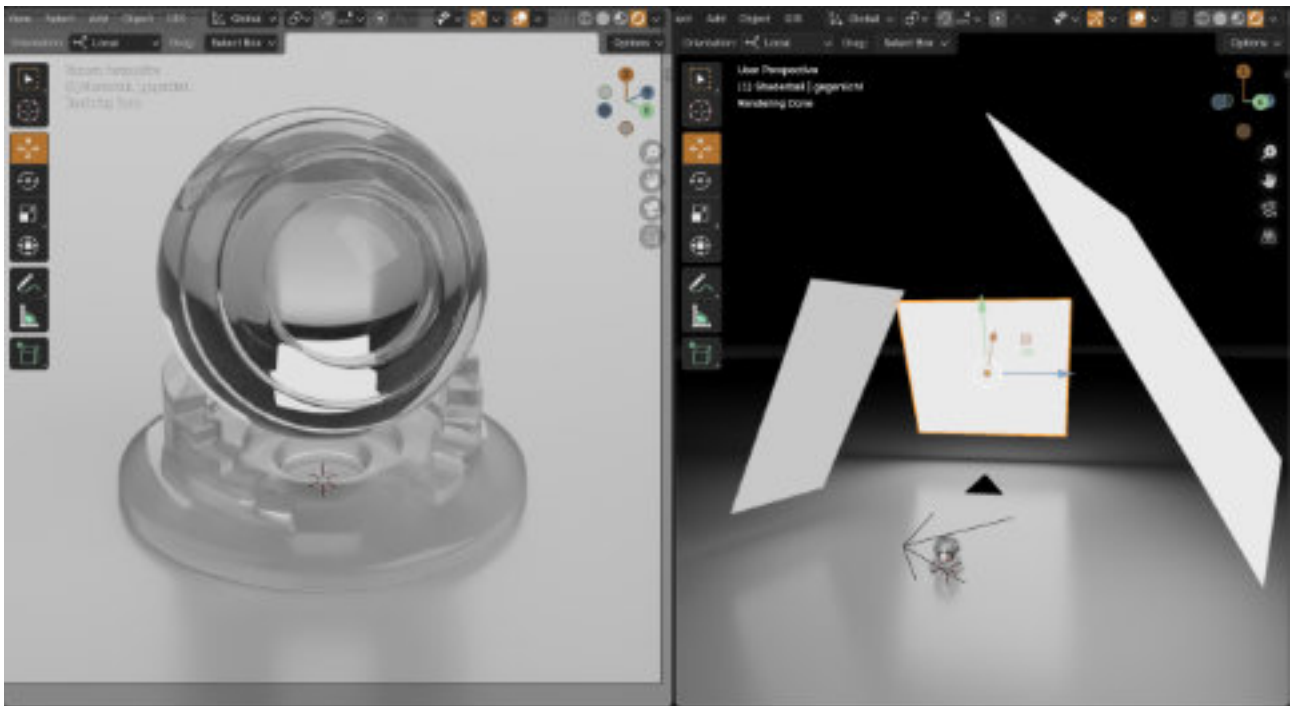


LIGHTMAPS

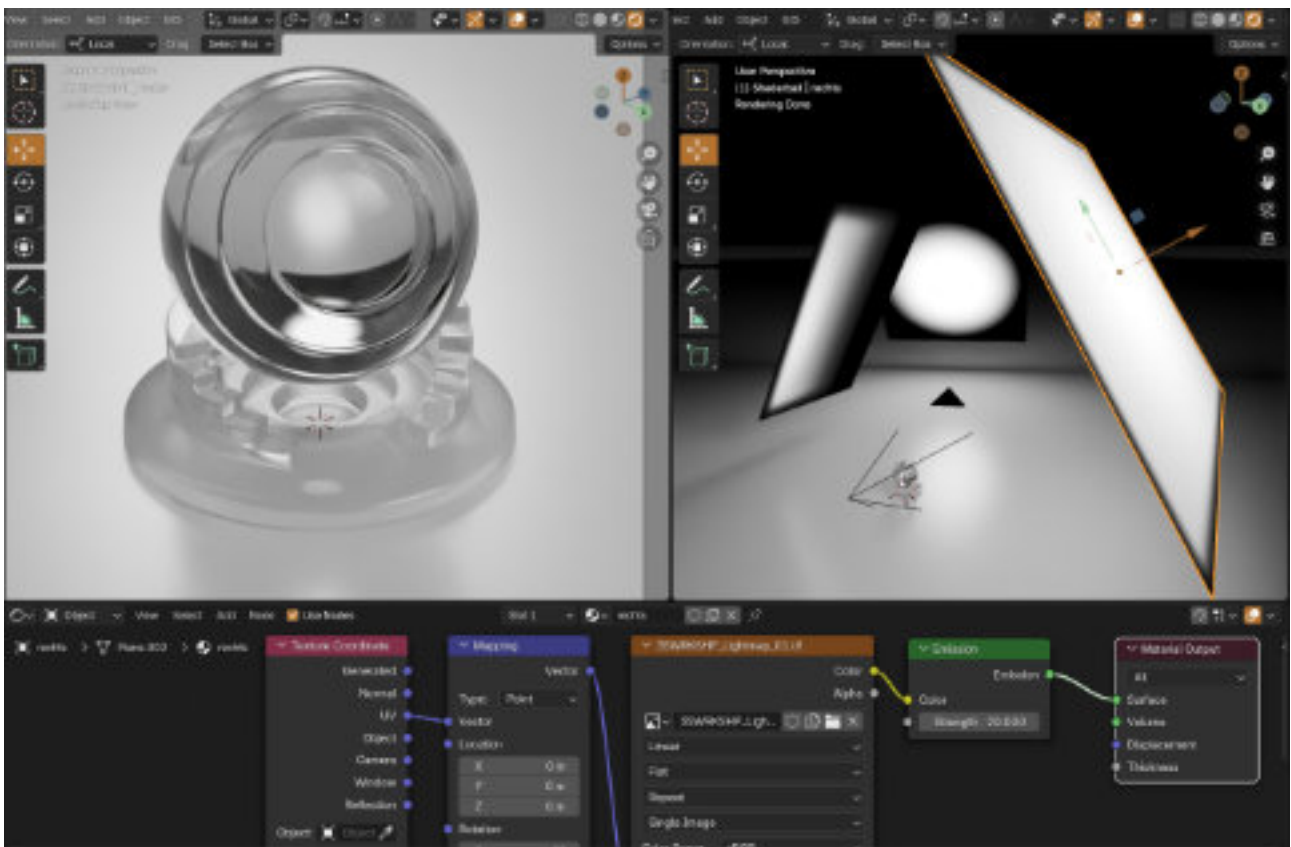
Gleichmäßig leuchtende Flächen erzeugen als Einspiegelung in glänzenden Oberflächen immer eine flache Beleuchtung. Eine plastische Materialwiedergabe lässt sich nur mit Verläufen erzielen, die in stark reflektierenden Oberflächen eingesiegelt werden.

Dies ist mit einem Arealight, wie es Blender zur Verfügung stellt nicht möglich, da die leuchtende Fläche keine Helligkeitsunterschiede aufweist. Um eine plastische Materialwiedergabe in glänzenden Oberflächen zu erreichen, eignen sich Lightmaps.





Beleuchtung mit gleichmäßig leuchtenden Flächen.



Beleuchtung mit Lightmaps

Die Verwendung von Lightmaps ist sowohl mit den Arealights, wie auch mit Planes, in Kombination mit einem Emitter möglich.

MODELS

3

HARDSURFACE-MODELING VS SOFTBODY MODELING

Hard Surface Modeling und Softbody Modeling sind zwei sehr unterschiedliche Ansätze im 3D-Modellieren und werden für verschiedene Arten von Objekten verwendet.

1. Hard Surface Modeling

Hard Surface Modeling wird verwendet, um Objekte zu erstellen, die überwiegend harte, kantige Oberflächen haben und typischerweise aus Materialien bestehen, die sich nicht verformen, wie Metall, Kunststoff oder Glas. Diese Technik eignet sich gut für technische, mechanische und architektonische Modelle.

Um klare Kanten und definierte Winkel zu erhalten, eignen sich am besten die klassischen Modellierungswerkzeuge wie Extrusion, Beveling, und Boolean Operations.

2. Softbody Modeling

Diese Modellierungstechnik wird verwendet um Objekte mit organischen, geschwungenen und weichen Formen zu erstellen. Als Basis für solche Objekte dienen oft niedrig aufgelöste 3D-Objekte, die mithilfe der oben genannten Werkzeuge erstellt und unter Verwendung eines Subdivision Surface Modifier abgerundet werden.

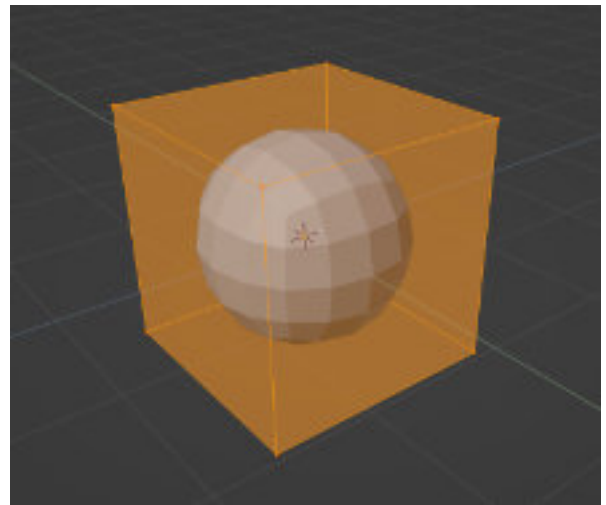
Eine weitere Technik um Objekte mit organischen Formen zu erstellen ist die Sculpting Technik. Objektgruppen bei denen das Softbody Modeling der ideale Umsetzungsweg ist, sind Objekte wie Kissen und Kleidung oder auch geschwungene Formteile aus Kunststoff und Metall..

SUBDIVISION MODIFIER

Der Subdivision Modifier teilt die vorhandenen Flächen eines Modells in kleinere Flächen auf. Und glättet dabei die Geometrie um dynamischere Übergänge zwischen den flachen des Objekts zu erreichen.

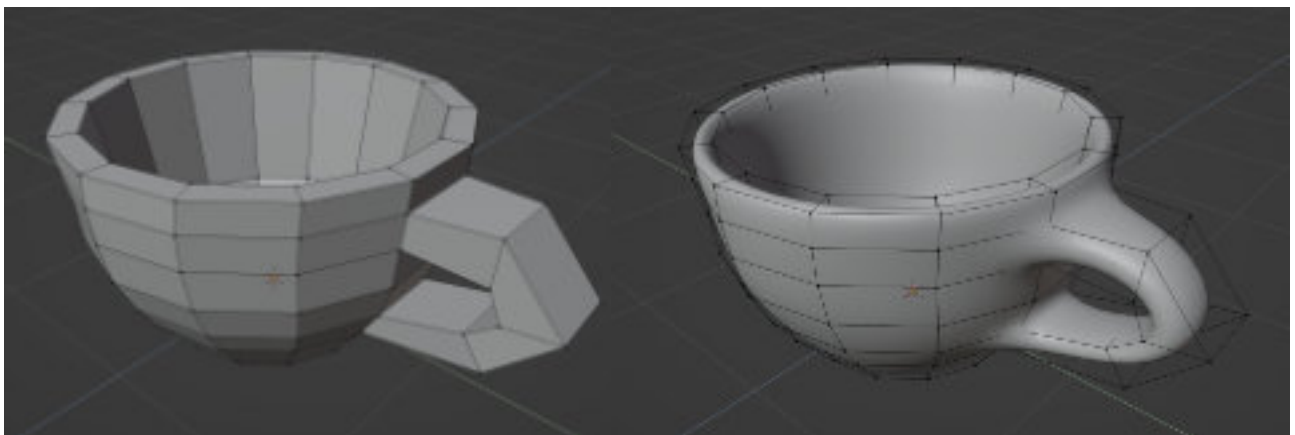
Er basiert auf Subdivision Surfaces, einem mathematischen Algorithmus, der von Catmull-Clark entwickelt wurde.

Jede Fläche des Modells wird in kleinere Flächen unterteilt. Dadurch wird das Modell glatter, weil die zusätzlichen Flächen basierend auf den benachbarten Punkten interpoliert werden.



Der Modifier arbeitet nicht zerstörend: Die ursprünglichen Geometriepunkte bleiben erhalten und können weiterhin bearbeitet werden. Die neue, feinere Geometrie wird dynamisch berechnet.

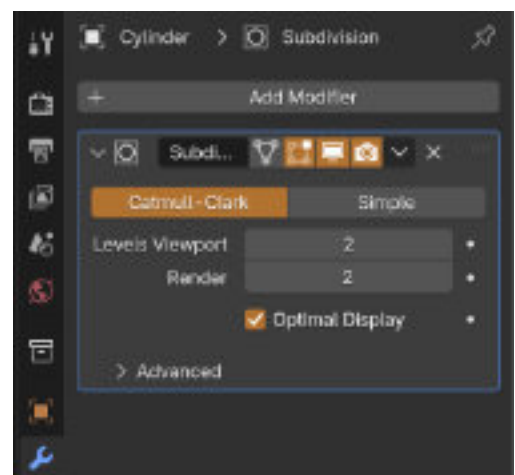
Ein Würfel, bestehend aus sechs Flächen wird mit einem Subdivision Modifier, durch die Abrundung von Kante zu Kante zu einer Kugel.



Der Subdivision Modifier ist am rechten Rand in den Properties in der Abteilung Modifier zu finden? Das ist der Reiter mit dem blauen Schraubenschlüssel.

Die Einstellung: „Levels Viewport“ legt die Anzahl der Unterteilungen für die Darstellung im 3D-Editor fest und die Einstellung „Render“ legt fest, wie fein das Objekt im tatsächlichen Rendering unterteilt wird.

Der Modifier lässt sich mit dem Befehl Apply auf das Objekt anwenden. Das Objekt hat dann die Geometrie, die durch den Modiefier simuliert wurde.



SHADER

4

SHADER

Unter Shader werden alle Eigenschaften zusammengefasst, die beeinflussen, wie sich Licht verhält, das auf die Oberfläche eines Objekts trifft. Lichtstrahlen können zum Beispiel reflektiert oder absorbiert werden.

Wenn Lichtstrahlen reflektiert werden kann es sich um eine diffuse Oberfläche handeln und die Lichtstrahlen werden in unterschiedliche Richtungen reflektiert. Ist eine Oberfläche Nichtdiffus, dann gilt für die Reflektion das Prinzip: Einfallswinkel gleich Ausfallswinkel.



Wenn sie nicht reflektiert werden, dann werden sie absorbiert. Die Lichtstrahlen bewegen sich nun durch ein Objekt hindurch. Auch für diesen Weg werden eine Menge an Informationen benötigt, um die Objekte und Oberflächen im Rendering realistisch dazustellen: In welchem Winkel setzt das Licht seinen Weg fort? (IOR) oder wurden die Lichtstrahlen auf dem Weg durch ein Objekt gestreut? (SSS SubsurfaceScattering).

Um physikalisch korrekte und optisch ansprechende Oberflächen zu erstellen können noch eine ganze Menge weiterer Informationen in den Shader einfließen, so ändert sich bei manchen Oberflächen das Reflektionsverhalten auch noch in Abhängigkeit vom Betrachtungswinkel (Fresnelleffekt).

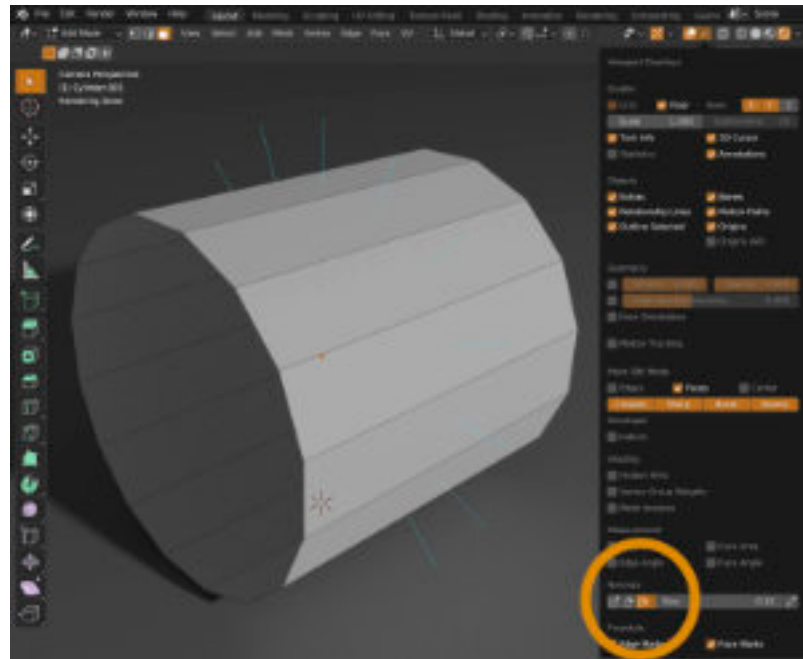
All diese Informationen zusammengekommen definieren was mit dem Licht beim Auftreffen auf eine Oberfläche passiert und werden unter dem Begriff Shading zusammengefasst.

NORMALS

Warum sind Normals wichtig?

Normals sind ein wesentlicher Bestandteil beim Shading, da sie die Ausrichtung einer Fläche oder eines Punktes im Raum definieren.

Eine Normal ist ein Vektor, der senkrecht (90°) zur Oberfläche einer 3D-Fläche oder eines Punktes steht. Normals helfen, die Richtung zu bestimmen, in die eine Fläche „zeigt“.



Im Shading haben Normals eine große Bedeutung. Die Ausrichtung einer Fläche ist entscheidend dafür, wie sich das auf die Fläche auftreffende Licht verhält. Je direkter eine Fläche zur Lichtquelle ausgerichtet ist, umso heller wird die Fläche im Rendering dargestellt. Je abgewandter sie von der Lichtquelle ist, umso dunkler wird sie nachher im Rendering sein.

Im Menü des Viewport Overlays lässt sich die Sichtbarkeit der Normals für den Viewport Ein- und Ausschalten.

Neben den Flächen-Normals gibt es auch noch Vertex-Normals, welche die Ausrichtung eines Punktes im Raum definieren.

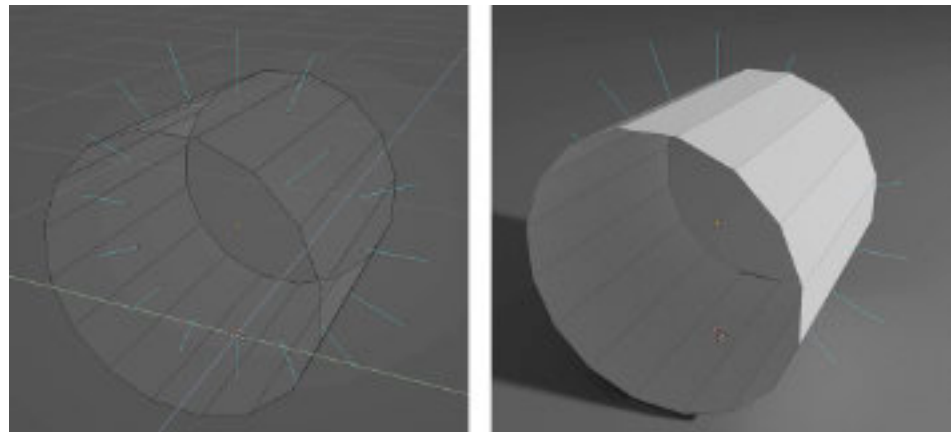
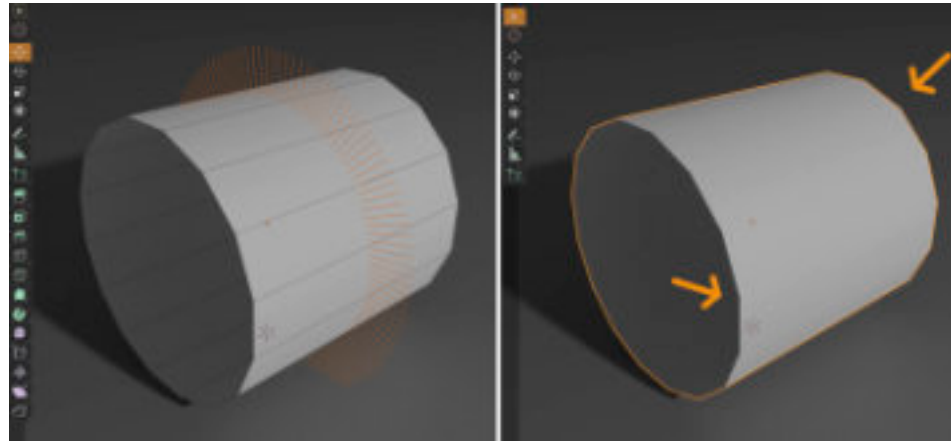
Shadingfehler entstehen oft durch falsch ausgerichtete Normals. Bei der Verwendung von Bump-, Normal- oder Displacement-Maps können falsch ausgerichtete Normals dafür sorgen, dass ein Relief nicht nach außen, sondern nach innen gewölbt dargestellt wird. Solche Fehler sind oft nicht auf den ersten Blick zu erkennen, produzieren aber eine unrealistische Darstellung der Oberfläche.

Flat Shading / Smooth Shading

Beim Flat-Shading ist jede Fläche eines Objekts erkennbar, da die einzelnen Flächen, durch die Polygonkanten, klar gegeneinander abgegrenzt sind. Dies wird im Shading dadurch erreicht, dass die normale, an jeder Stelle der Fläche, die gleiche Ausrichtung hat: rechtwinklig zur Fläche.

Beim Smooth-Shading sind die Kanten der Flächen nicht mehr zu sehen. Es entsteht eine gleichmäßige Fläche mit weichen Übergängen an den Polygonkanten. Für die gleichmäßige Darstellung der Oberfläche werden über die komplette Fläche interpolierte Normalen simuliert, die sich zur Kante der Fläche immer mehr nach der Normale der Nachbarfläche ausrichten.

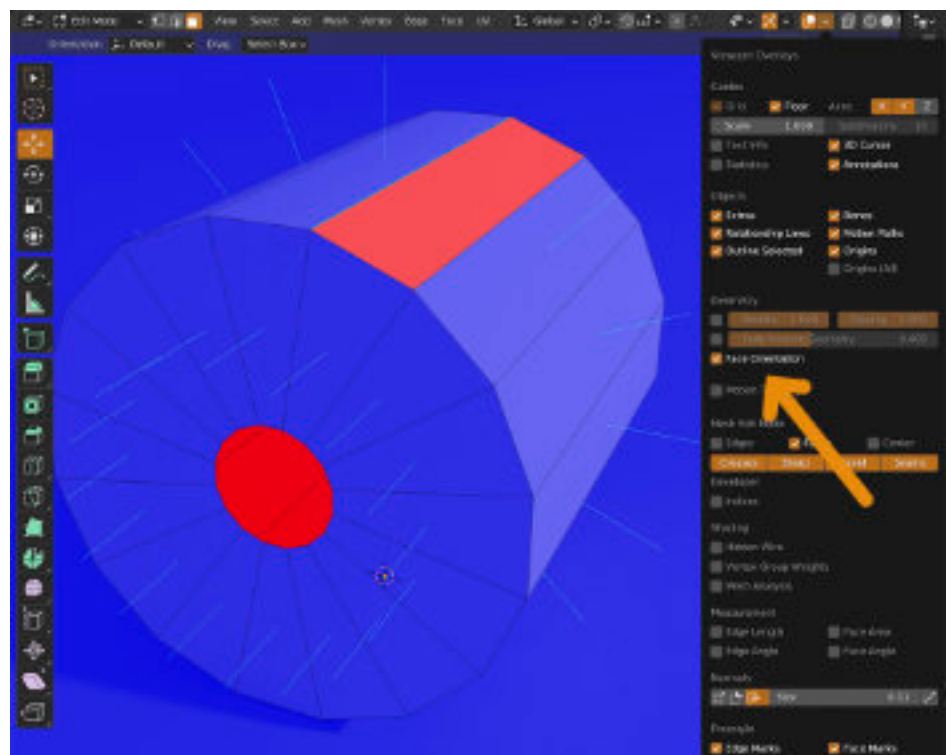
Im oberen Bild ist gut zu sehen, dass durch das Shading, zwar eine gleichmäßig abgerundete Oberfläche simuliert wird, die Auflösung des Objekts aber nach wie vor so gering ist, dass an der Objektkante ganz deutliche Abstufungen zu sehen sind.



Ausrichtung der Normalen

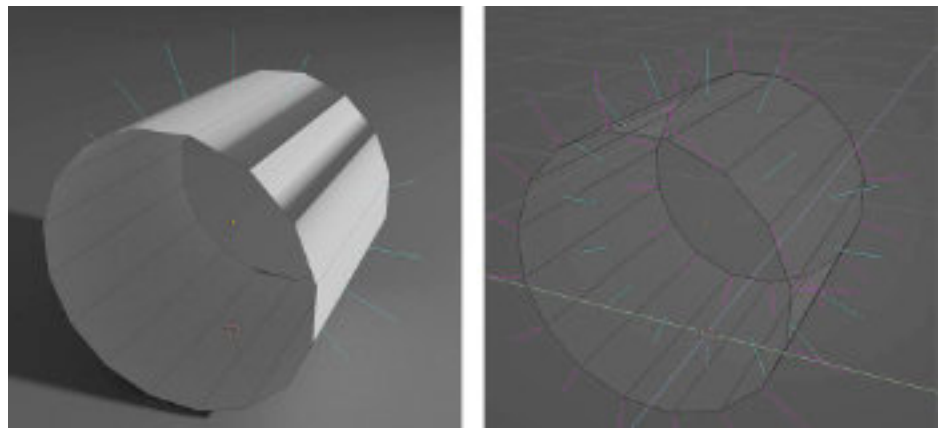
Für ein sauberes Shading ist die Ausrichtung der Flächen mit ihren dazugehörigen Normalen wichtig. Beim oben abgebildeten Objekt zeigt eine Normale nach innen. Das macht beim Flat-Shading keinen Unterschied, was daran liegt, dass bei Blender beide Seiten einer Fläche sichtbar sind. Nicht alle Renderengines und einige Game-Engines machen beide Seiten einer Fläche sichtbar. Hier würde schon im Flat-Shading ein Fehler auftauchen.

In den Viewport Overlays lässt sich die Ausrichtung aller Flächen direkt sichtbar machen. Alle Flächen die

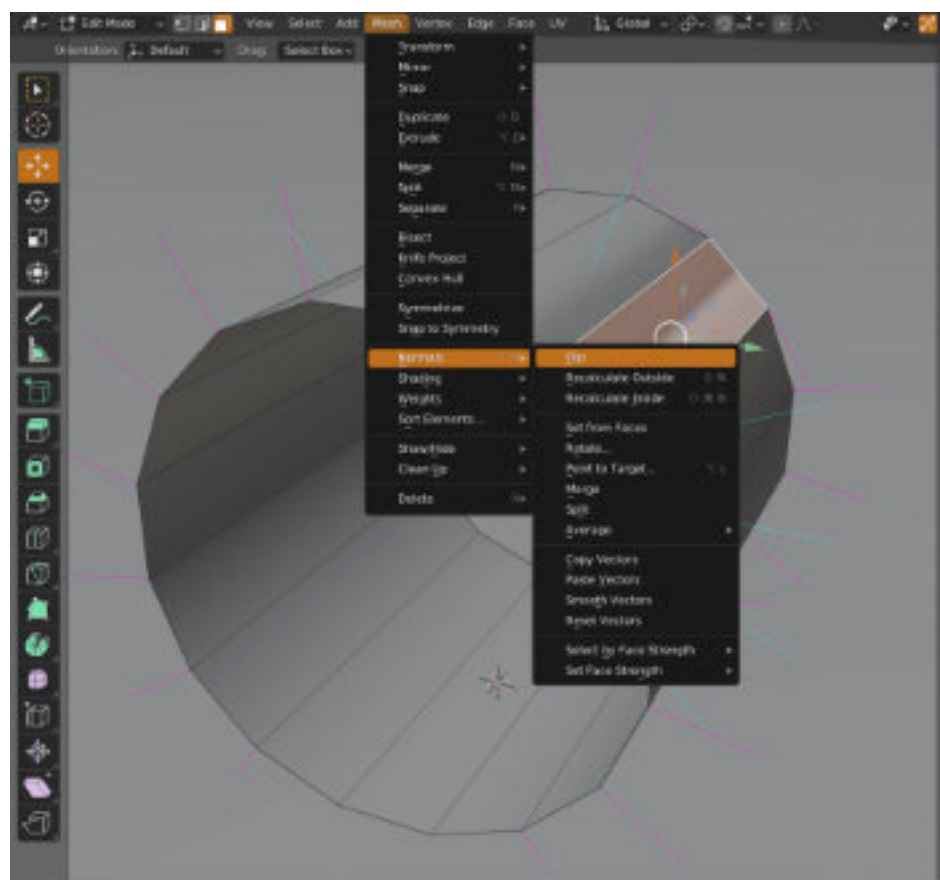


nach Außen zeigen sind blau eingefärbt und alle nach innen gerichteten Flächen sind rot.

Beim **Smooth-Shading** führt die umgedrehte Fläche allerdings zu einem Shadingfehler. Die interpolierten Normalen an der Polygonkante liegen parallel zur Fläche, da die Normale der Nachbarfläche nach innen zeigt.

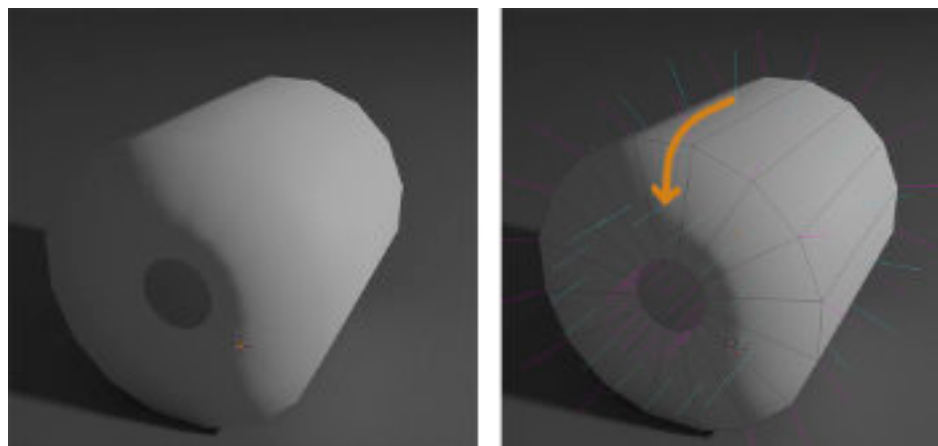


Um dies zu beheben lassen sich im Edit-Mode Flächen auswählen und unter Mesh > Normals > Flip Normals die Ausrichtung der Flächen und ihrer Normalen ändern.



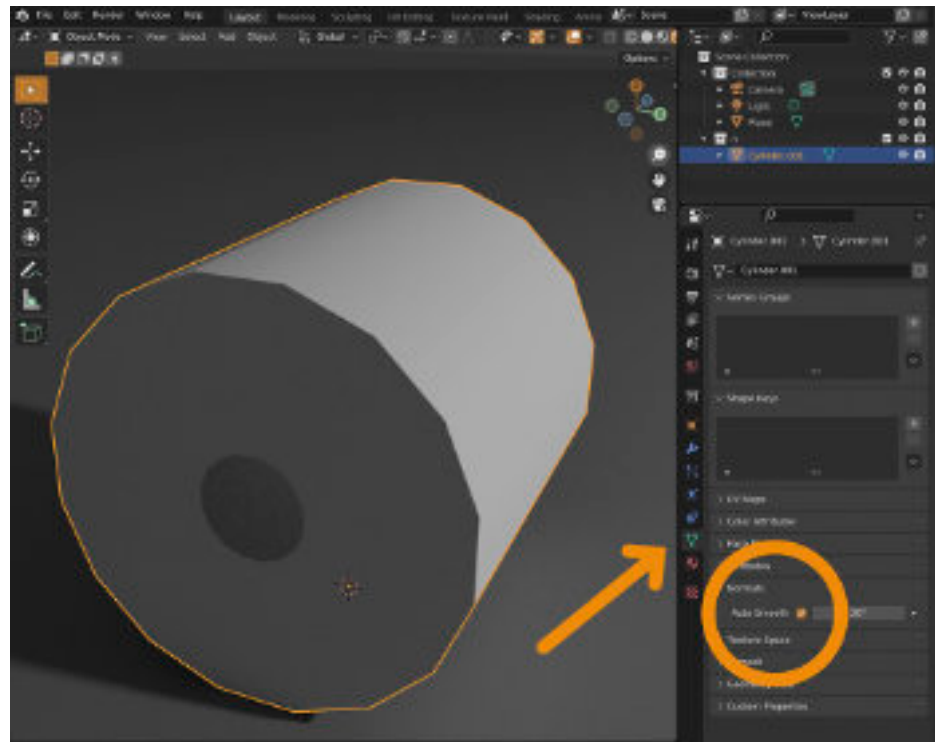
Autosmooth und Hard-Edges

Wenn mein Beispielobjekt nun auch noch Flächen auf den Stirnseiten bekommt, geht es mit den Shadingfehlern gleich weiter. Bei aktiviertem Smooth-Shading wird die Darstellung nun auch über die Außenkanten abgerundet.



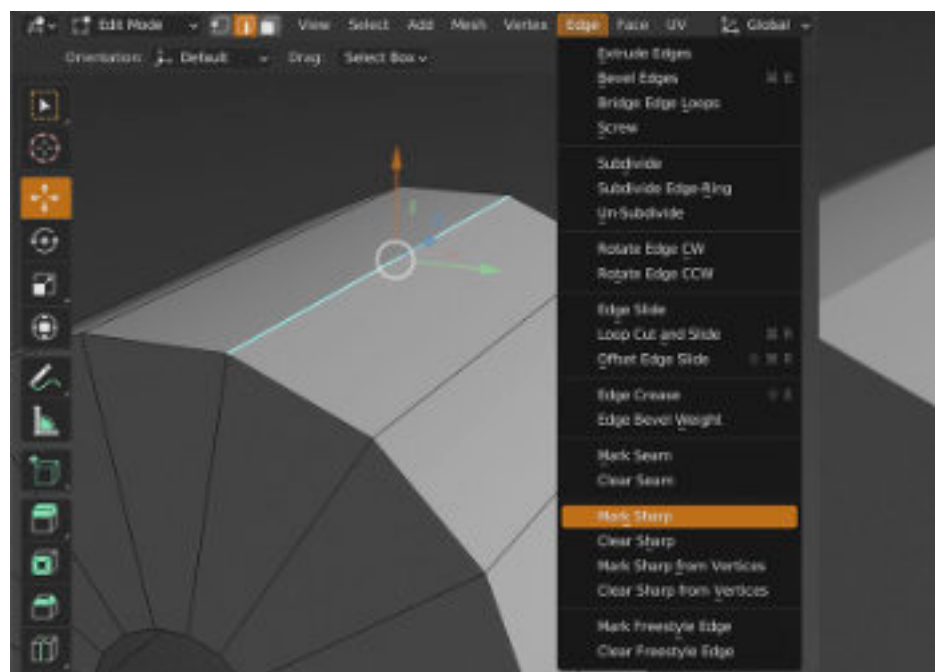
Dies lässt sich mit Autosmooth verhindern. Die Funktion ist in den Scene-Properties, im Vertex Tab, unter Normals versteckt. Für Autosmooth lässt sich eine Winkelbegrenzung angeben, die verhindert dass Flächen, welche in einem steileren Winkel, als dem angegebenen, aneinanderstoßen, abgerundet dargestellt werden.

In diesem Fall liegt die Winkelbegrenzung bei 30° . Da die umlaufenden Flächen in einem stumpferen Winkel aneinander stoßen, wird der Zylinder abgerundet dargestellt. Der Winkel zu den Deckflächen liegt bei 90° , ist somit spitzer als 30° und es entsteht eine harte Kante.



Zusätzlich lassen sich Kanten, von der Abrundung ausschließen, auch wenn sie innerhalb der Winkelbegrenzung liegen. Dazu müssen die Kanten selektiert und im Edit Mode unter: Edge > Mark Sharp markiert werden.

Es ist nur möglich Kanten von der Abrundung im Smooth-Shading auszuschließen, wenn Autosmooth aktiviert ist!



PBR WORKFLOW

Der PBR-Workflow(Physically Based Rendering) ist eine Methode in der 3D-Grafik, die darauf abzielt, Materialien und Oberflächen so realistisch wie möglich darzustellen, indem sie physikalische Prinzipien der Lichtinteraktion simuliert. Licht wird von Oberflächen reflektiert, gestreut oder absorbiert. Der PBR-Workflow versucht dies nach den physikalischen Regeln so genau wie möglich nachzubilden.

Die beiden Haupt-PBR-Workflows sind der Metall-Roughness-Workflow und der Specular-Glossiness-Workflow. Blender verwendet den ersten der beiden genannten Workflows.

Die grundlegenden Oberflächeneigenschaften werden durch folgende Parameter bestimmt: Metallic und Roughness. Hierbei Gibt Metallic an ob ein Material metallisch ist oder nicht. Roughness steuert die Rauheit der Oberfläche und beeinflusst, wie stark das Licht gestreut oder reflektiert wird.

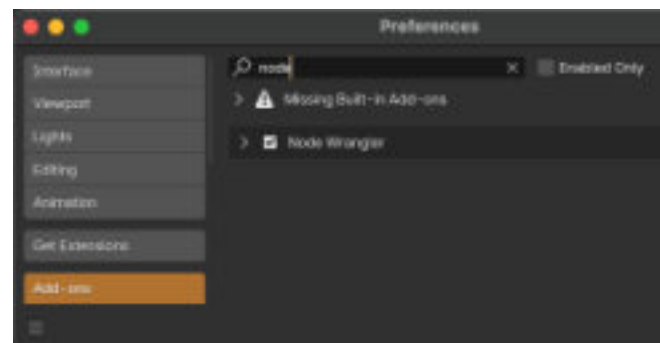
NODE WRANGLER

Um Texturen mit einem Shader zu verknüpfen, sie genau zu positionieren und diese nachher skalieren und drehen zu können, werden drei Nodes benötigt: ein Image Texture Node, ein Texture Coordinate Node und ein Mapping Node.

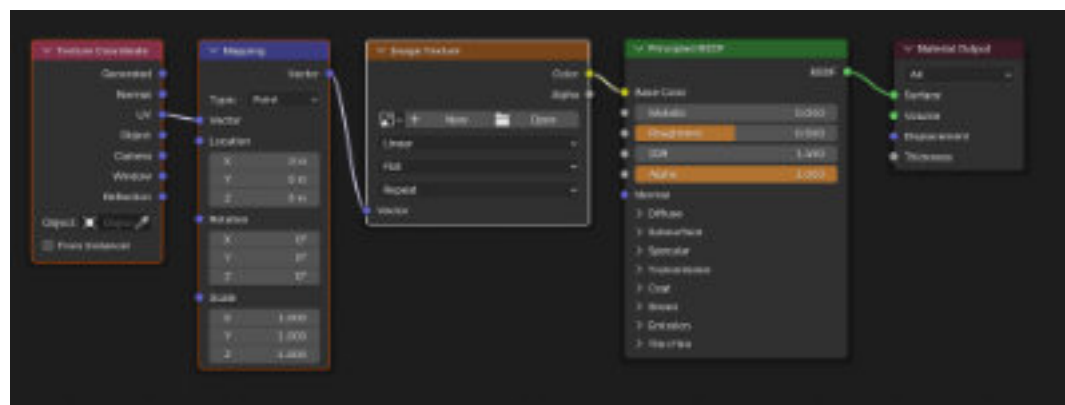
Node Wrangler ist ein Add On, das diese Nodes mit einem Shortcuts erstellt und direkt verknüpft.

Node Wrangler muß nicht installiert werden. Er ist in der Blender Installation bereits enthalten und muß lediglich aktiviert werden.

Hierzu in den Preferences, unter Edit auf Add Ins gehen und im Suchfeld Node Wrangler eingeben. Wenn noch kein Häkchen gesetzt ist dann ist er noch nicht aktiviert. Nach der Aktivierung funktioniert das Erstellen der oben genannten Nodes mit einem Klick.



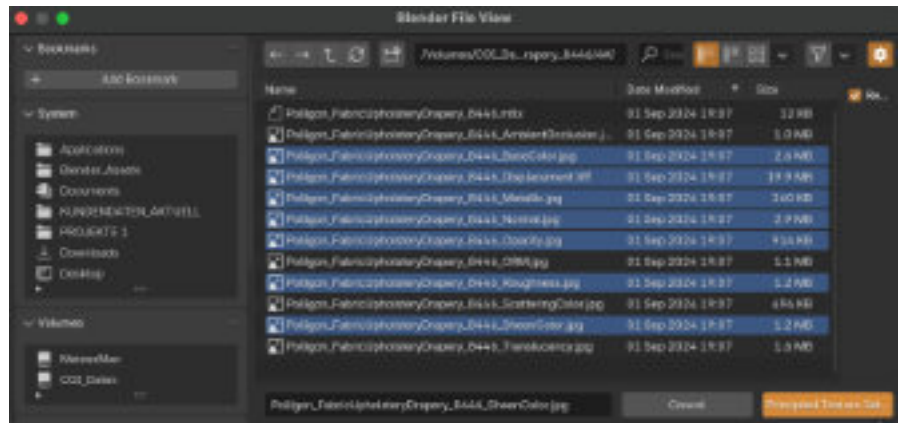
Im Shader Editor muß der Shader, für den die Nodes erstellt werden sollen angewählt sein. Mit der Tastenkombination CTRL T werden die beiden Nodes erzeugt und korrekt verknüpft.



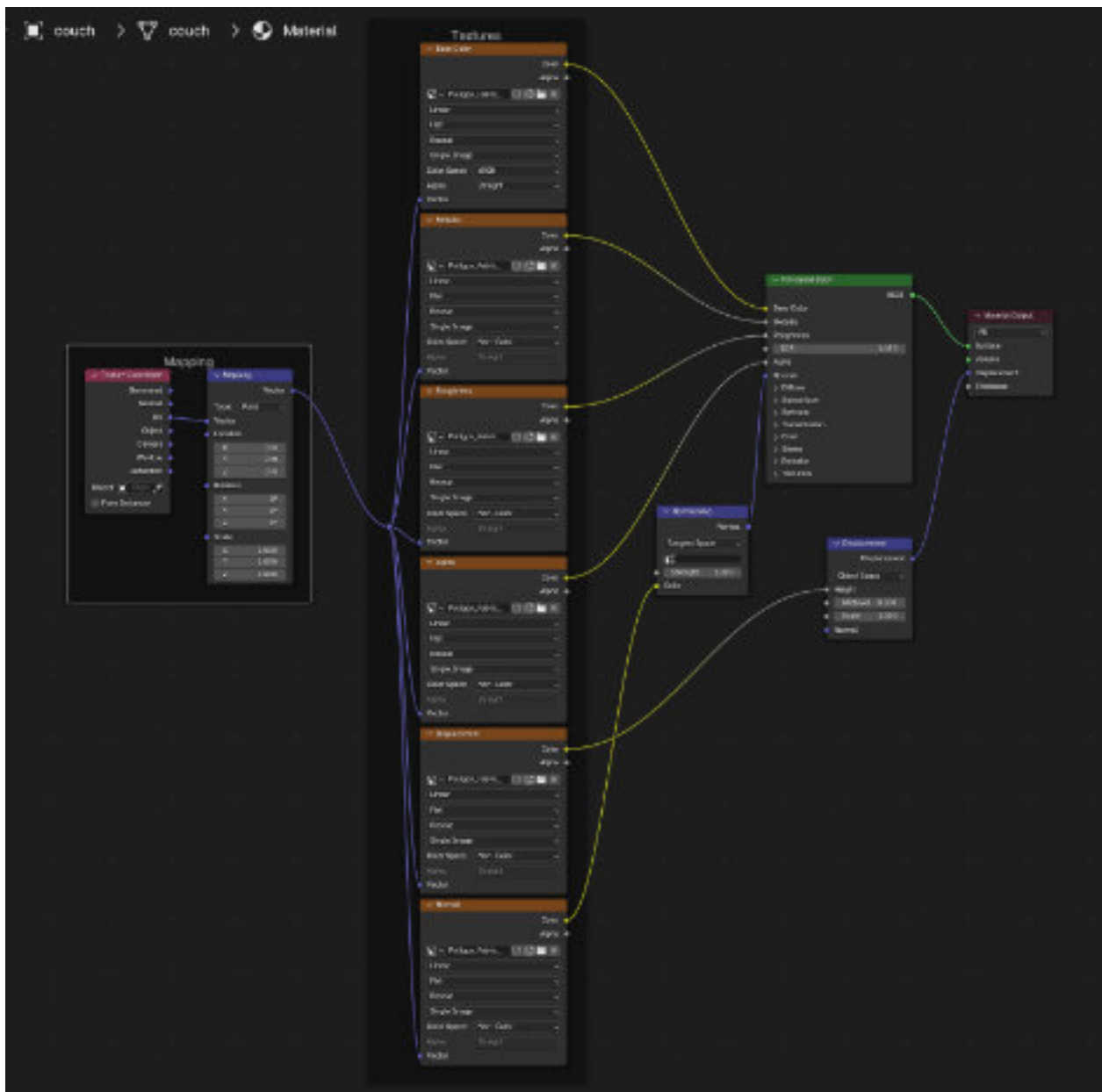
Mit dem Shortcuts Shift CTRL T erstellt Node Wrangler ein komplettes

Node Setup nach dem PBR-Workflow. Nach drücken von Shift CTRL T erscheint ein Fenster in dem nach dem Ordner der Maps gefragt wird. In dem gewünschten Ordner müssen nun alle

erforderlichen Maps (Albedo bzw. Base Color), Roughness, Metallic, Bump Map, Normal Map und Displacement Map ausgewählt werden. Nach Bestätigung der Auswahl erstellt Node Wrangler das komplette Node-Setup mit allen benötigten Verknüpfungen.



Die Werte für die jeweiligen Maps müssen allerdings noch mit Testrenderings ermittelt werden.



BSDF SHADER

Der BSDF (Bidirectional Surface Distribution Function) Shader ist der vielseitigste unter den Shadern in Blender. In diesem Shader sind die wichtigsten Materialeigenschaften zusammengefasst um völlig unterschiedliche Materialien zu erstellen.

Die wichtigsten Materialkanäle:

Base Color: Definiert die Oberflächenfarbe

Metallic: Wert 0 = nicht metallische Oberfläche / Wert 1 = metallische Oberfläche

Roughness: Wert 0 = glatte Oberfläche / Wert 1 = diffuse Oberfläche

IOR (Index of Refraction): Brechungsindex

Alpha: In den Alpha-Kanal lässt sich eine Maske laden, die das Objekt an abmaskierten Stellen unsichtbar macht.

Subsurface: Legt fest wie die Lichtstrahlen gestreut werden, wenn sie nicht reflektiert werden und in das Material eindringen. Mehr unter SSS

Specular: Regelt die Reflektionsintensität

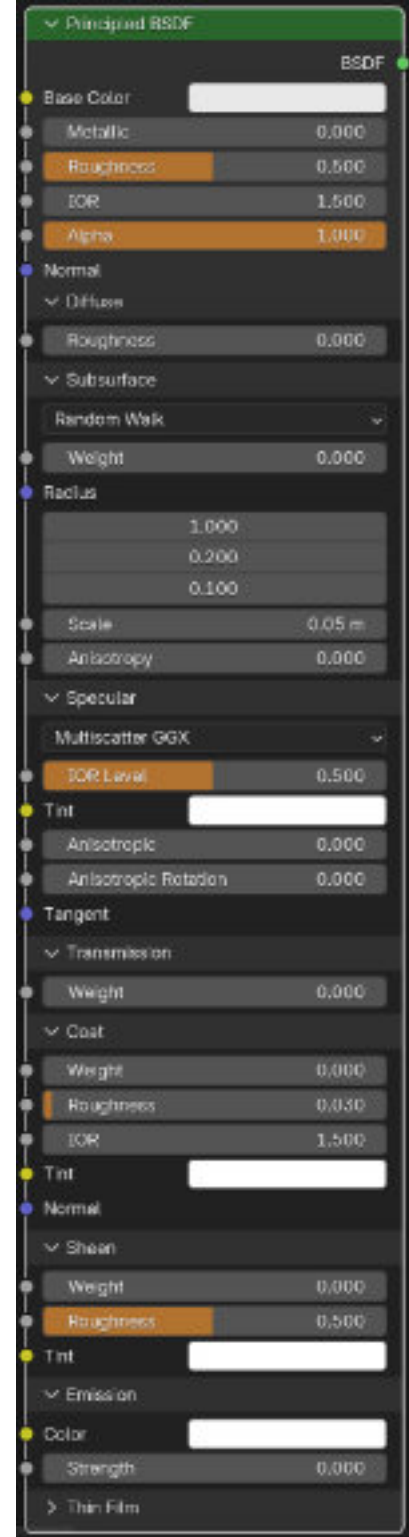
Transmission: Ist für die Transparenz zuständig

Coat: Bietet die Möglichkeit eine zusätzliche Lackschicht auf die Oberfläche aufzutragen.

Sheen: Sorgt für einen zusätzlichen Glanzeffekt bei extremen Reflexionswinkeln. Eignet sich gut für Samt

Emission: macht aus dem BSDF-Shader einen Emission Shader, der die Oberfläche des Objekts leuchten lässt.

Thin Film:



NEW ROUGHNESS IN BLENDER 4.3

Der Roughness Kanal im BSDF-Shader von Blender wurde in der Version 4.3 komplett überarbeitet.

Ist eine Oberfläche **nicht** Diffus verläuft die Berechnung der Reflektierten Lichtstrahlen nach dem Prinzip: Einfallswinkel gleich Ausfallswinkel.

Handelt es sich um eine diffuse Oberfläche werden auftreffende Lichtstrahlen in unterschiedliche Richtungen reflektiert.

In den Versionen **vor** 4.3 wurde eine diffuse Oberfläche nach dem Lambertschen Modell berechnet. In der neuen Version steht zusätzlich zur Lambertschen Methode, die Berechnung von diffusen Oberflächen nach der Oren-Nayar Methode zur Verfügung.

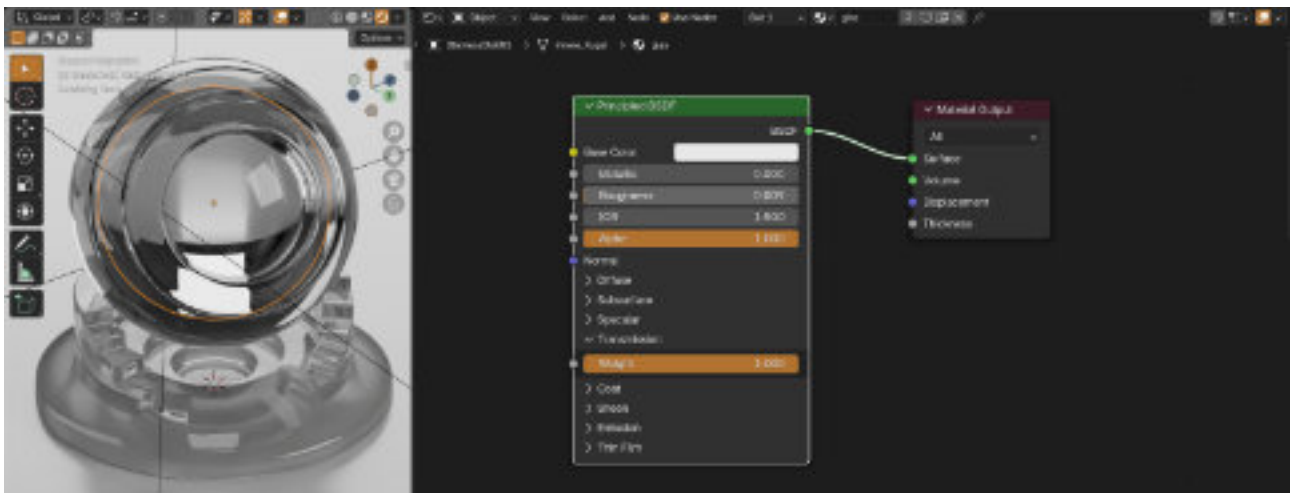
Das Lambert Modell besagt dass Lichtstrahlen, die auf eine diffuse Oberfläche treffen in zufälliger Richtung reflektiert werden.

Das Oren-Nayar-Modell geht hier den physikalisch wesentlich präziseren Weg. Dieses Simulationsmodell geht davon aus, dass eine Oberfläche nicht perfekt glatt sein kann und immer kleine Unebenheiten aufweist. Das Oren-Nayar Modell simuliert eine zufällige Verteilung kleiner Unebenheiten auf der Oberfläche und berechnet dann auf Basis dieser Unebenheiten die exakten Richtungsänderungen der reflektierten Lichtstrahlen.

Auf den ersten Blick ist der Unterschied zwischen den beiden Simulationsmodellen nicht zu sehen. Bei genauer Betrachten wirken die Schattenbereiche lebendiger und weisen wesentlich komplexere Aufhellungen durch Gegenreflexionen auf.

GLAS SHADER

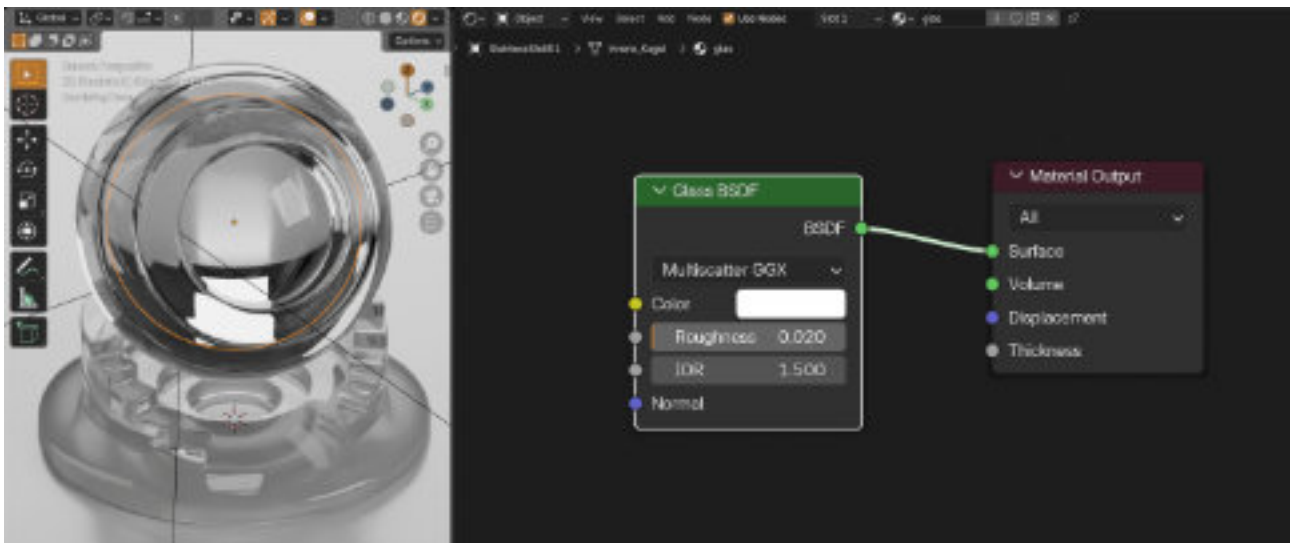
Die einfachste Art ein lichtdurchlässiges Material zu erstellen, ist die Verwendung des BSDF Shaders und unter „Transmission“ den Weight-Wert zu erhöhen. Durch das Erhöhen des Wertes wird das Material immer lichtdurchlässiger. Beim maximalen Wert von 1,0, erhält man ein transparentes Material welches wie Glas aussieht.



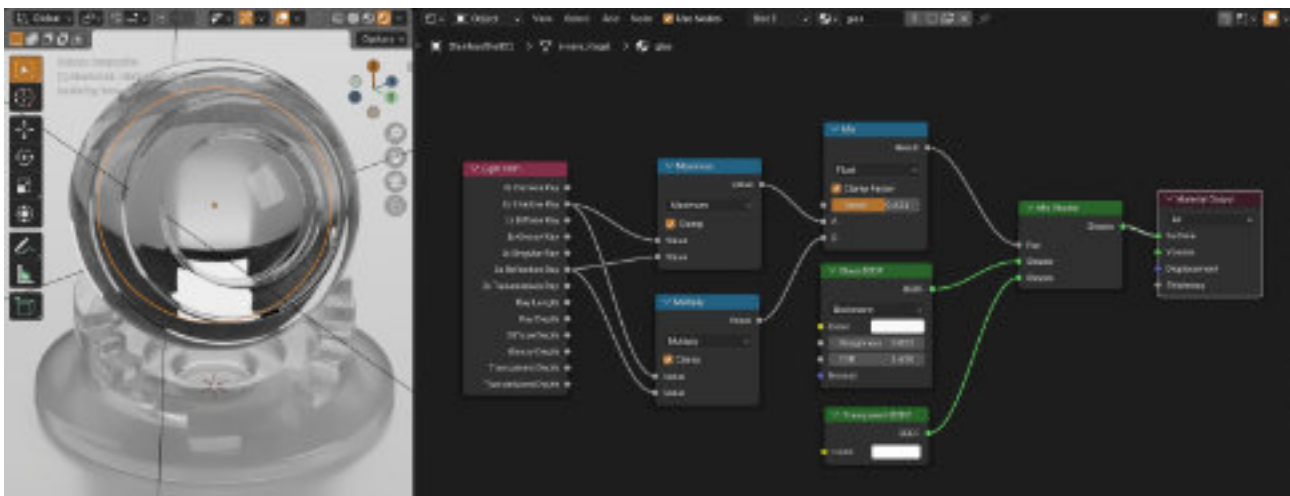
Über „Base-Color“ lässt sich die Färbung des Glases einstellen und eine Veränderung des „Roughness“-Parameters definiert die Beschaffenheit der Oberfläche. Ein Wert über 0 lässt die Glasoberfläche immer mehr wie satiniertes Glas erscheinen, je höher der Wert ist.

Da ein Lichtstrahl, der in ein Objekt eindringt immer die Richtung ändert, ist der IOR-Wert (Index of refraction) bei transparenten Materialien wichtig. Unterschiedliche Glassorten haben unterschiedliche Brechungswinkel und Wasser hat wieder einen anderen Brechungsindex als Glas.

Hierfür gibt es Tabellen mit den genauen IOR-Werten für die jeweiligen transparenten Materialien.



Eine weitere Möglichkeit ein transparentes Material zu erstellen, ist der Glas-Shader. Diesem Shader fehlen Parameter, die im BSDF-Shader vorhanden sind, aber in einem transparenten Material nicht benötigt werden. Die Berechnung des Shaders im Rendering unterscheidet sich auch von einem transparenten Material auf Basis des BSDF-Shaders.



Der realistischste mir bekannte Glas-Shader baut auf einem recht komplexen Node Setup auf. Ich habe einen Screenshot des vollständigen Shaders hier abgedruckt.

SUB SURFACE SCATTERING

Rechenintensiv wird es wenn Licht nicht nur in ein Material eintritt, sondern wenn es unter der Oberfläche diffus gestreut wird und an anderer Stelle wieder austritt.

Anders als bei vollkommen undurchsichtigen Materialien wird das Licht also nicht nur von der Oberfläche reflektiert, sondern durchdringt die obere Schicht des Materials und wird dabei diffus verteilt.

Benötigt werden diese Shader für Materialien wie Haut, Milch, Wachs und diverse gummiartige Kunststoffe.

Wie funktioniert Sub Surface Scattering?

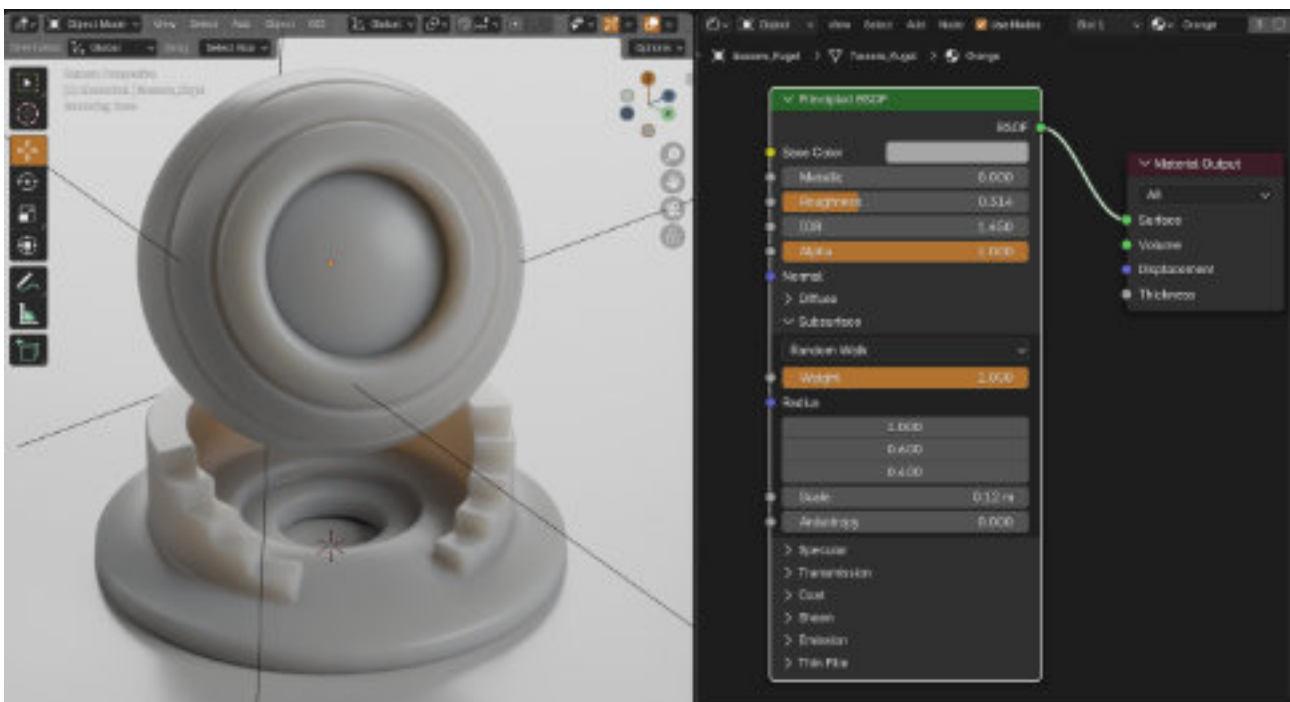
Einfallendes Licht:

1. Einfallendes Licht trifft auf die Oberfläche eines Materials.
2. Das Licht dringt in das Material ein und wird in alle Richtungen gestreut. Die Tiefe der Streuung hängt von den Materialeigenschaften ab.

Austritt des Lichts:

3. Nach der Streuung tritt ein Teil des Lichts wieder aus der Oberfläche aus. Dies verleiht dem Material ein weiches, leicht durchscheinendes Aussehen.

Blender bietet eine einfache Möglichkeit, SSS zu nutzen, besonders über den Principled BSDF Shader.



Die wichtigsten Parameter:

WEIGHT:

Ein Wert von 0 bedeutet kein Subsurface Scattering (vollständig undurchsichtig). Höhere Werte aktivieren den Effekt und erhöhen die Lichtstreuung unter der Oberfläche.

RADIUS:

Bestimmt, wie tief das Licht in das Material eindringen kann. Jeder Farbkanal (R, G, B) kann separat angepasst werden, um unterschiedliche Streutiefen für verschiedene Farben zu simulieren.

SCALE:

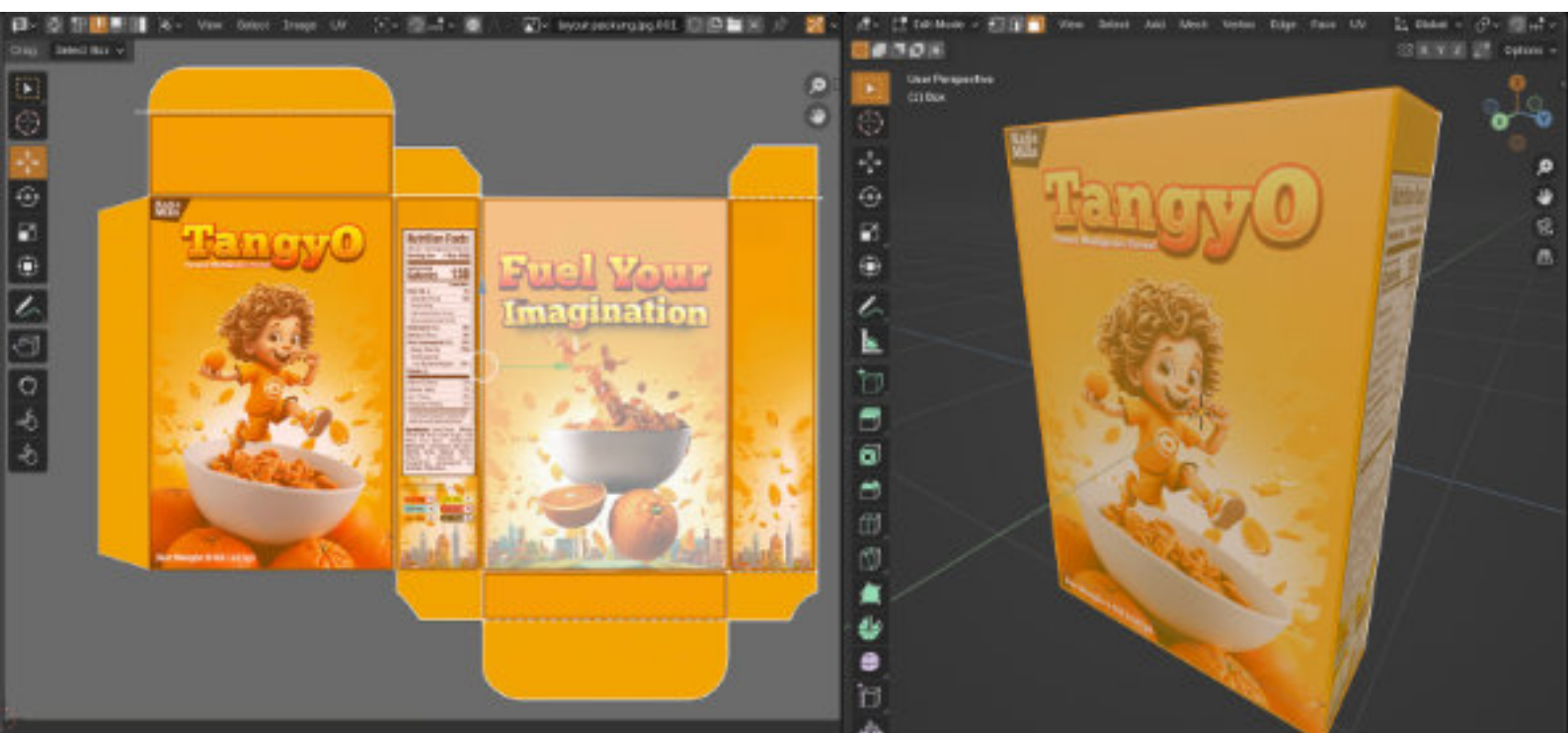
Legt die generelle Eindringtiefe des Lichts fest. Unabhängig von der Wellenlänge.

UV-MAPPING

Das UV-Mapping ist ein grundlegendes Verfahren, um zweidimensionale Texturen korrekt auf dreidimensionale Modelle anzuwenden.

Die Textur ist wie ein flaches Bild (z. B. eine JPG-Datei).

Über UV-Koordinaten wird definiert, welcher Bereich der Textur auf welche Fläche des Modells projiziert wird. Die UV-Koordinaten sind ein 2D-Koordinatensystem (U und V anstelle von X und Y), das bestimmt, wie eine Textur auf ein 3D-Modell „geklebt“ wird.



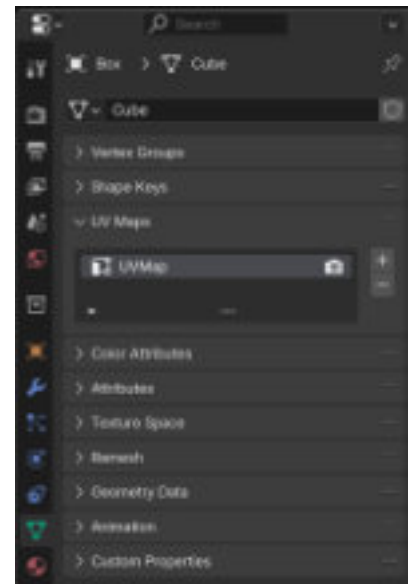
Die Verwendung der beiden Buchstaben U und V ist willkürlich, wichtig ist nur dass sie nicht die Buchstaben X, Y oder Z enthalten, da diese in der 3D-Welt zwar die drei Richtungsachsen bezeichnen aber nicht genormt sind. Beispiel: In Blender ist die Z Achse immer die Vertikale Achse. In Cinema 4D ist es die X Achse.

Um eine Textur auf ein 3D-Objekt aufzubringen, wird das Modell virtuell „auseinandergeschnitten“ und in einer 2D-Fläche abgewickelt. Dieser Prozess wird oft mit dem Entfalten eines Papiermodells verglichen. Die entfaltete Form nennt man die UV-Map. Jede Oberfläche des 3D-Modells wird einem Bereich der UV-Map zugeordnet. Die Textur auf der UV-Map wird dann auf die entsprechenden Bereiche des Modells projiziert.

Die Schwierigkeit beim UV-Mapping darin, ein dreidimensionales Objekt möglichst verzerrungsfrei in eine zweidimensionale Fläche abzuwickeln.

Die Vorgehensweise im UV-Mapping macht wie folgt Sinn

1. Wechsel in die Arbeitsumgebung des UV-Editors.
2. Das Objekt sollte sich im Edit Mode befinden und alle Flächen sollten ausgewählt sein.
3. Wenn es noch keine UV-Map gibt sollte in den Vertex-Properties, unter UV-Maps eine UV-Map erstellt werden.
4. Nicht nur im 3D-Viewport, auch im UV-Editor müssen alle Flächen ausgewählt sein, bevor das Unwrapping gestartet wird. Der Unwrapping Prozess bezieht sich nur auf selektierte Flächen.
5. Unwrapping starten
6. Flächen sortieren und neu ausrichten.

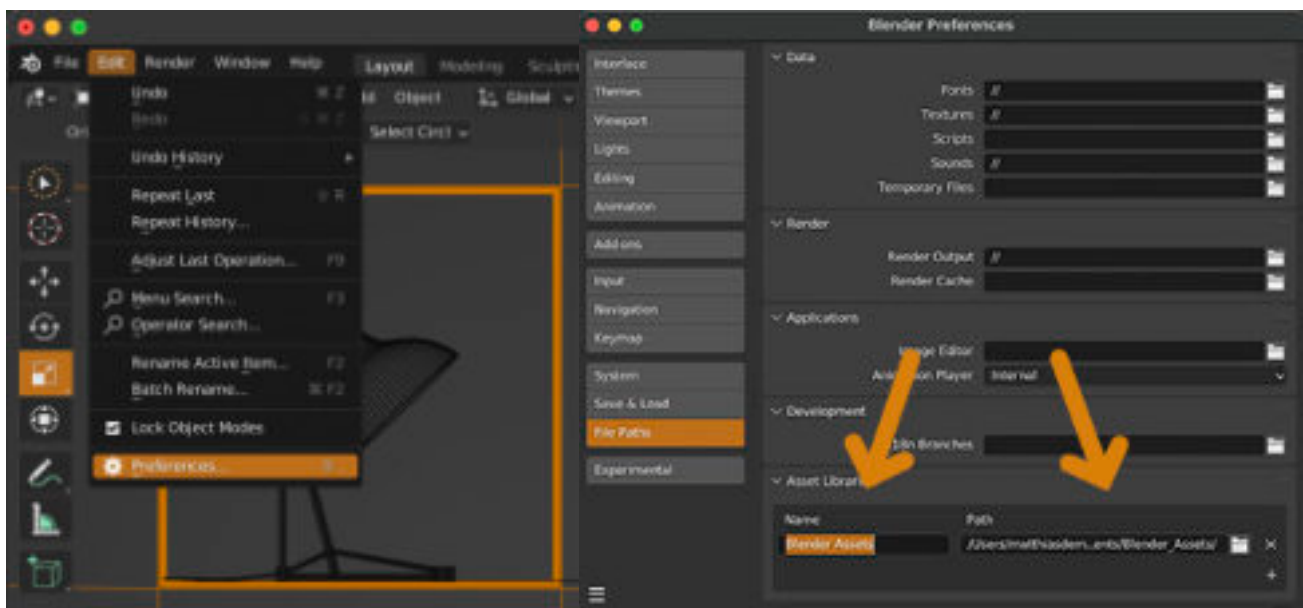


ASSETS

5

ASSET MANAGER

Der Asset Manager bietet eine komfortable Möglichkeit eine Vielzahl von Assets wie 3D-Objekte und Texturen übersichtlich zu verwalten. Es ist möglich mehrere Bibliotheken an unterschiedlichen Orten zu verwalten, aus denen die jeweiligen Dateien per Drag and Drop direkt in Blender übernommen werden können.

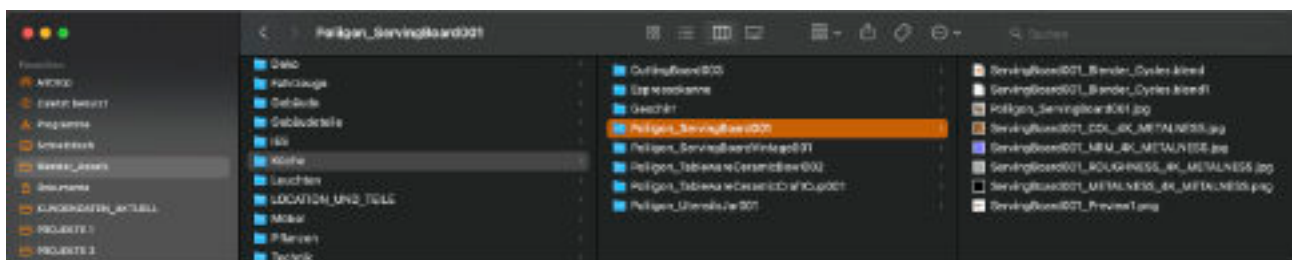


Zuerst muß hierfür ein Bibliotheksordner erstellt und dieser mit Blender verknüpft werden. Die Verknüpfung findet unter „Edit“ in den Preferences statt. In den Voreinstellungen gibt es am unteren Ende der Menüs den Eintrag „File Paths“

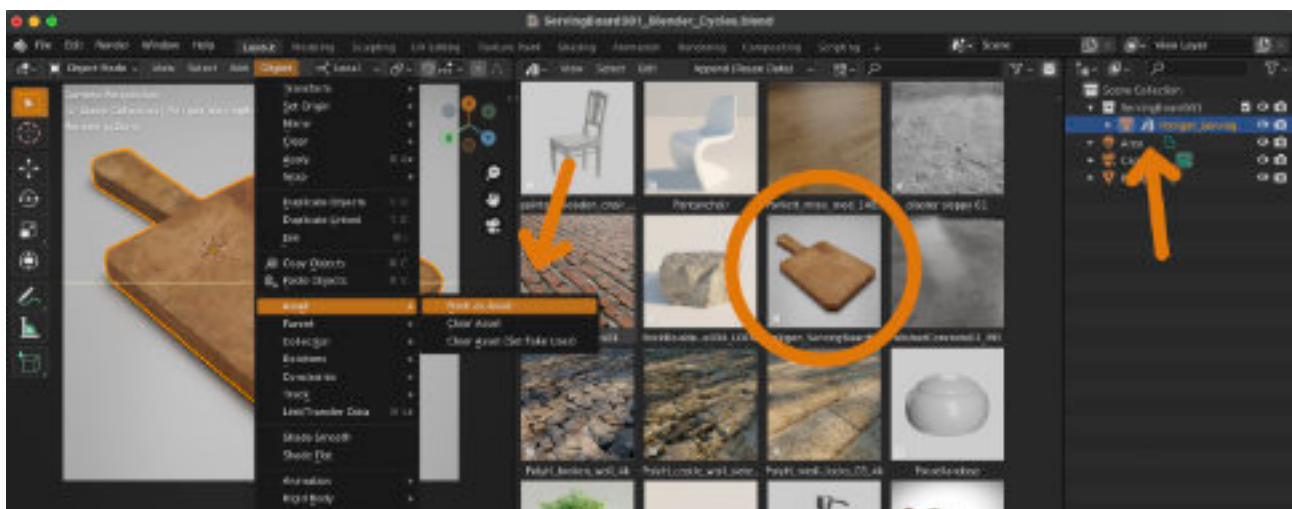
Dort wird unter Asset Libraries über das + Symbol eine neue Library erstellt. Unter Path sollte ein Speicherort gewählt werden, auf den Blender immer zugreifen kann. Wenn sich die Bibliothek auf einer externen Festplatte befindet muß sichergestellt sein, dass diese bei der Arbeit mit Blenderdateien, die Objekte oder Texturen aus der Bibliothek verwenden, immer verfügbar ist.

Dateien die Assets enthalten, die in der Library organisiert sind sollten inklusive Texturen und verknüpfter Dateien im eben erstellten Library Ordner abgelegt werden. Innerhalb dieses Ordners sind auch Unterordner möglich.

Darüberhinaus können auch mehrere Objekte aus einem Blenderfile als eigenständige Assets markiert werden.

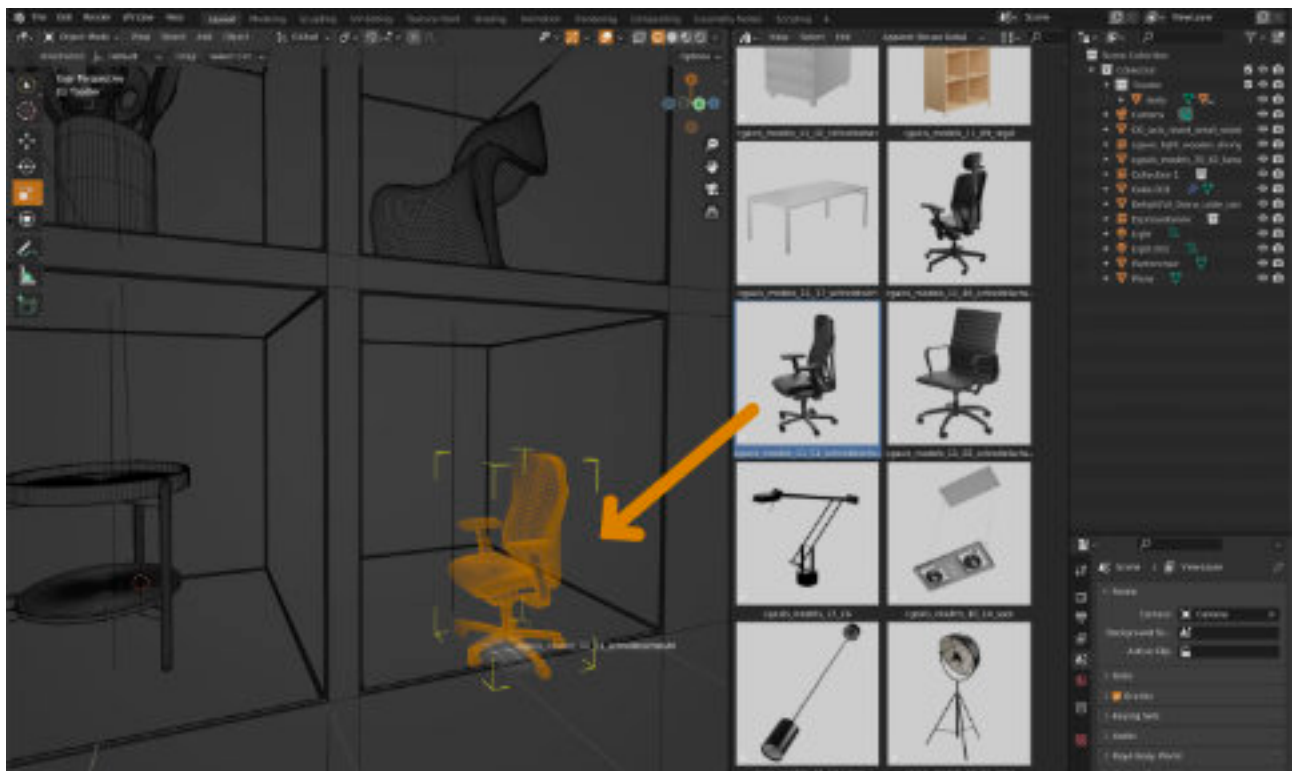


Um ein Objekt in der Asset-Library zur Verfügung zu stellen, muß es im Outliner als Asset markiert werden. Das Objekt steht nun in der Bibliothek zur Verfügung.



Zugriff auf die verfügbaren Librarys erhält man über den Asset-Browser. Im linken oberen Bereich lassen sich durch ein Drop-Down Menü die unterschiedlichen Bibliotheken auswählen.

Für eine bessere Übersichtlichkeit lassen sich den einzelnen Assets individuelle Vorschaubilder zuordnen. Hierfür muß das gewünschte Asset ausgewählt werden und über das Sidepanel im Asset Browser (wird mit der Taste N geöffnet) unter Preview ein Bild ausgewählt, das sinnvollerweise dort liegt, wo auch die Blenderdatei mit den verknüpften Assets abgelegt ist.



WOCHENPLAN

MONTAG

Rendersettings
realTime compositor
rendering ID Masks
non destruktives Arbeiten

kurze Wiederholung Kamera positionieren / selektive schärfe

ÜBUNG: Frühstückstisch
Lichtsetting: Gogo: Plane > Noise texture > color ramp
Lightmaps
IES Files

DIENSTAG

Hardsurface Modeling / Softbody Modeling

ÜBUNGEN: Cappucinotasse / Milchkanne / Milchflasche mit Milch
Lenorflasche

MITTWOCH

UV Mapping
UV Unwrapping

ÜBUNGEN Müslipackung Texturieren
Tasse oder Kanne Texturieren
Waschmittelflasche Texturieren

DONNERSTAG

Shader / was ist Shading
BSDF Shader (neues Roughness)
Transparente Shader
Glas Shader
SSS
PBR Workflow

ÜBUNGEN Milchflasche mit Milch
Teetasse mit Tee

FREITAG

Sculpting

ÜBUNG Kissen und Sofa

Lightlinking / Lightgroups / Lightmaps / Assetbrowser