



Master 2 ISICG

Projet Géométrie Discrète et Images

Jalon 2 : Miroirs et transparence

Réalisé par :

BOURENANNE Younes Nacer Eddine

TRABELSI Ayoub

YAHYAOUI Mohamed Ala

Enseignant :

Monsieur MENEVEAUX Daniel

2023-2024

Table des matières

1	Introduction	1
2	Méthode de Calcul :	1
2.1	Miroir Parfait :	1
2.2	Facteurs de Réflexion de Fresnel :	2
2.3	Gestion de la Transmission :	2
3	Discussion	2
4	Résultats	3
5	Conclusion	6
6	Bibliographie	7

1 Introduction

Ce projet vise à développer un outil de visualisation 3D pour la représentation réaliste d'objets en utilisant le modèle de Cook-Torrance. À partir d'un canevas initial comprenant du code HTML et Javascript, nous nous concentrons initialement sur l'amélioration de l'outil existant.

Notre première étape consiste à enrichir cette plateforme par une interface HTML interactive pour contrôler les paramètres d'une source lumineuse ponctuelle, ainsi une manipulation pour analyser les effets sur l'apparence des objets 3D. L'objectif est d'intégrer au modèle existant le modèle de Cook-Torrance.

Notre démarche vise à élargir cette approche en introduisant un calcul d'éclairement basé sur une carte d'environnement. À partir de la carte d'environnement, nous avons utilisé la formulation des lois de Snell-Descartes et le facteur de Fresnel pour représenter des miroirs et des surfaces transparentes.

2 Méthode de Calcul :

La première étape consiste à définir une méthode de calcul de réflexion et de transmission dans le fragment shader. Pour ce faire, on a utilisé les fonctions **Reflect** et **Refract** fournies par GLSL, qui facilitent le calcul des directions réfléchies et transmises du rayon lumineux en fonction des lois de Snell-Descartes.

L'équation de Snell-Descartes pour la réfraction de la lumière à une interface entre deux milieux est définie comme suit :

$$\eta_i \sin \theta_i = \eta_t \sin \theta_t$$

Où :

- η_i est l'indice de réfraction du milieu incident,
- θ_i est l'angle d'incidence par rapport à la normale,
- η_t est l'indice de réfraction du milieu de transmission,
- θ_t est l'angle de transmission par rapport à la normale.

2.1 Miroir Parfait :

On a simplifié le modèle en considérant un miroir parfait. La réflexion devrait être parfaitement spéculaire, sans flou ni dispersion. On a réalisé des tests en utilisant une sphère parfaite, facilitant la vérification visuelle des réflexions. Les résultats étaient conformes aux attentes, montrant des réflexions nettes sur la sphère parfaite.

2.2 Facteurs de Réflexion de Fresnel :

On a introduit les facteurs de réflexion de Fresnel pour prendre en compte les variations de réflexion en fonction de l'angle d'incidence. Ceci a ajouté un réalisme supplémentaire en ajustant la force de réflexion en fonction de l'angle de vue. Les facteurs de réflexion **Rs** et **Rp** ont été utilisés pour la composante spéculaire, considérant la polarisation de la lumière.

Pour une lumière non polarisée, les facteurs de réflexion sont définis comme suit :

$$R_s = \left| \frac{\eta_i \cos \theta_i - \eta_t \cos \theta_t}{\eta_i \cos \theta_i + \eta_t \cos \theta_t} \right|^2$$

$$R_p = \left| \frac{\eta_i \cos \theta_t - \eta_t \cos \theta_i}{\eta_i \cos \theta_t + \eta_t \cos \theta_i} \right|^2$$

La conservation de l'énergie est maintenue avec $R = \frac{R_s + R_p}{2}$, et les proportions de transmission sont calculées avec $T = 1 - R$, $T_s = 1 - R_s$, et $T_p = 1 - R_p$.

2.3 Gestion de la Transmission :

Enfin, on a implémenté la gestion de la transmission, permettant à la lumière de passer à travers la surface. Cela a été réalisé en calculant la direction transmise du rayon lumineux à l'aide de la fonction `refract` et en respectant les lois de Snell-Descartes. Les surfaces transparentes montrent une transmission réaliste de la lumière à travers la sphère transparente.

3 Discussion

Pour l'implémentation de la carte environnement et des effets miroir et transparence dans notre programme, nous avons suivi le pseudocode suivant dans le fragment shader :

Nous avons initialisé l'indice de réfraction du milieu incident à 1.0 (air). Ensuite, nous avons calculé le cosinus de l'angle d'incidence et le cosinus de l'angle de transmission.

Ainsi, nous devons définir les six fonctions suivantes :

- `calculateRs` : fonction pour calculer **Rs** (réflexion spéculaire pour lumière non polarisée)
- `calculateRp` : fonction pour calculer **Rp** (réflexion spéculaire pour lumière non polarisée)
- `calculateReflectance` : fonction pour calculer **R** (réflexion totale)
- `calculateTs` : fonction pour calculer **Ts** (transmission spéculaire pour lumière non polarisée)

- calculateTp : fonction pour calculer $\mathbf{T_p}$ (transmission spéculaire pour lumière non polarisée)
- calculateT : fonction pour calculer \mathbf{T} (transmission totale)

Ensuite, nous avons calculé $\mathbf{R_s}$, $\mathbf{R_p}$, \mathbf{R} , $\mathbf{T_s}$, $\mathbf{T_p}$, et \mathbf{T} . Ensuite, nous avons calculé la couleur réfléchie et transmise à l'aide des fonctions prédéfinies reflect et retract.

Une des difficultés que nous avons rencontrées lors de la programmation était que, lors de la rotation de l'objet avec la souris, il affichait toujours la même face de la carte environnement sur l'objet. Pour remédier à cela, nous avons tenté de régler le problème en ajustant manuellement la position de la caméra. Cependant, le résultat n'était pas très satisfaisant.

4 Résultats

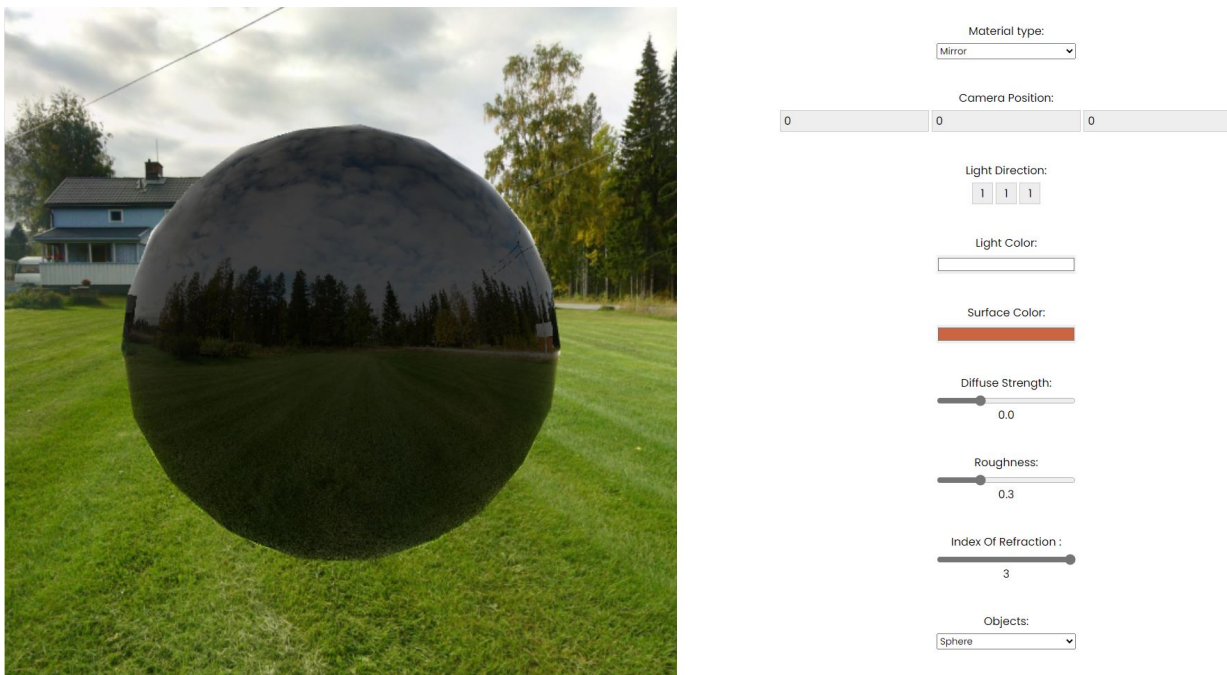


FIGURE 1 – Résultat 1



Material type:

Camera Position:

Light Direction:

Light Color:

Surface Color:

Diffuse Strength:

Roughness:

Index Of Refraction :

Objects:

FIGURE 2 – Résultat 2



Material type:

Camera Position:

Light Direction:

Light Color:

Surface Color:

Diffuse Strength:

Roughness:

Index Of Refraction :

Objects:

FIGURE 3 – Résultat 3



Material type:
Mirror

Camera Position:
0 0 0

Light Direction:
1 1 1

Light Color:
[Color Picker]

Surface Color:
[Color Picker]

Diffuse Strength:
0.0

Roughness:
0.3

Index Of Refraction :
3

Objects:
Bunny

FIGURE 4 – Résultat 4



Material type:
Transparent

Camera Position:
0 0 0

Light Direction:
1 1 1

Light Color:
[Color Picker]

Surface Color:
[Color Picker]

Diffuse Strength:
0.0

Roughness:
0.3

Index Of Refraction :
1

Objects:
Bunny

FIGURE 5 – Résultat 5

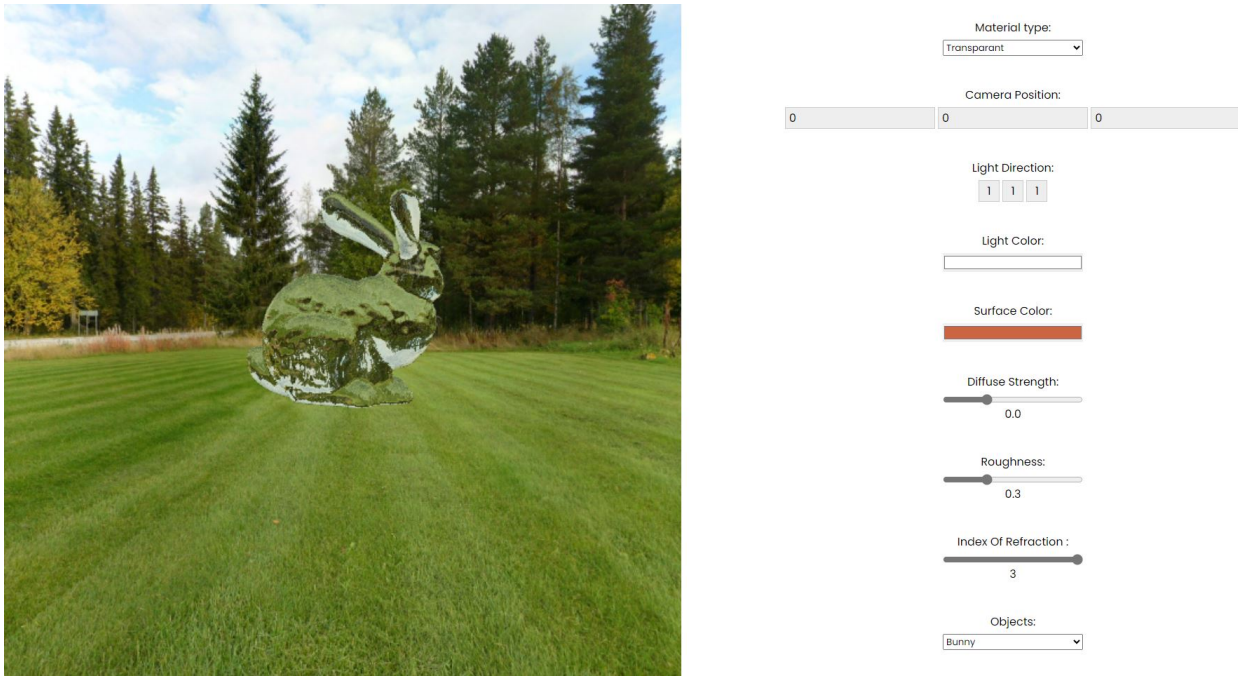


FIGURE 6 – Résultat 6

5 Conclusion

En conclusion, cet outil de visualisation des objets en 3D nous a permis de mieux découvrir le modèle de Cook-Torrance ainsi que l'intégration de cartes d'environnement et la simulation de miroirs et de surfaces transparentes.

Pour la réalisation de ce travail, nous avons utilisé les langages HTML et JavaScript avec le canvas OpenGL. Nous avons réussi à implémenter le modèle Cook-Torrance en ajoutant des variables pour représenter la source lumineuse, sa puissance, ainsi que les paramètres de réflectance de la surface, puis nous avons ajouté une carte d'environnement.

Aussi, nous avons réussi à implémenter les lois de Snell-Descartes et le facteur de Fresnel pour représenter des surfaces réfléchissantes et transparentes.

6 Bibliographie

Ci-dessous une collection de références vers les différentes ressources nous ayant servi lors de la réalisation de ce projet.

Références

- [1] *R.L. Cook and K.E. Torrance, A Reflectance Model for Computer Graphics, ACM Transactions on Graphics, 1982*

<https://webglfundamentals.org/webgl/lessons/webgl-skybox.html>

<https://opengameart.org/content/field-skyboxes>