



UNIVERSITE ABDELMALEK ESSAADI  
FACULTE DES SCIENCES ET TECHNIQUES DE TANGER  
DÉPARTEMENT GENIE INFORMATIQUE

Année universitaire 2024-2025



---

## Mini-Projet 2 : Développement d'un Système d'Indexation et de Recherche par Contenu dans une Base de Modèles 3D

---

**Réalisé par :** ESSETTI Widad  
EL GHAZI Loubna  
ELAOUFI Imane

**Encadré par :** Pr. M'hamed Ait Kbir

Année Universitaire : 2024/2025

## Table des matières

1.	Introduction .....	3
2.	Problématique .....	3
3.	Objectifs .....	3
4.	Conception et Développement.....	4
4.1.	Architecture .....	4
4.2.	Diagramme de classe.....	5
5.	Implémentation des méthodes pour le calcul des descripteurs.....	5
5.1.	Voxelisation.....	5
5.2.	Transformée de Fourier 3D .....	6
5.3.	Extraction des coefficients significatifs .....	6
5.4.	Moments de Zernike .....	6
5.5.	Normalisation des descripteurs.....	7
6.	Évaluation de similarité par descripteurs.....	7
6.1.	Distance entre descripteurs.....	7
6.2.	Score global de similarité.....	8
7.	Réduction du maillage.....	8
7.1.	Décimation Quadratique.....	8
7.2.	Simplification des faces .....	8
8.	Réalisation.....	9
9.	Comparaison des Résultats .....	11
10.	Conclusion .....	12
11.	Webographie.....	13

# 1.Introduction

Avec l'explosion des données numériques en trois dimensions (3D) dans divers domaines tels que la conception assistée par ordinateur (CAO), les jeux vidéo, la médecine et les applications de réalité virtuelle, la nécessité de développer des systèmes efficaces d'indexation et de recherche devient cruciale. Ces systèmes permettent de localiser rapidement des modèles 3D pertinents à partir de grandes bases de données, en fonction de leurs caractéristiques géométriques et structurelles.

Cependant, la représentation et la recherche efficaces dans de telles bases posent des défis complexes, notamment en termes d'invariance à la rotation, à la translation et au changement d'échelle des modèles 3D. Ces problématiques appellent à l'utilisation de descripteurs robustes et compacts, capables de capturer l'essence des formes tout en facilitant leur comparaison.

## 2.Problématique

Avec l'augmentation des bases de données de modèles 3D, la recherche efficace et précise de formes similaires devient un défi majeur, surtout face aux transformations géométriques telles que la rotation, la translation et le changement d'échelle. Les descripteurs de forme, tels que les coefficients de Fourier et les moments de Zernike, offrent des solutions prometteuses grâce à leur capacité à représenter les modèles dans des espaces compacts et invariants.

Ce projet cherche à répondre aux questions suivantes :

1. Quelle est l'efficacité des descripteurs pour capturer la complexité des modèles 3D ?
2. Quels sont leurs avantages et leurs limites pour la recherche de similarité ?
3. Comment la réduction de maillage impacte-t-elle la qualité des descripteurs ?

L'objectif est de développer un système robuste et performant pour l'indexation et la recherche par contenu dans une base de modèles 3D, tout en évaluant les performances des descripteurs implémentés.

## 3.Objectifs

L'objectif principal de ce projet est de concevoir un système performant d'indexation et de recherche dans une base de modèles 3D, en utilisant des descripteurs de forme robustes et invariants aux transformations géométriques.

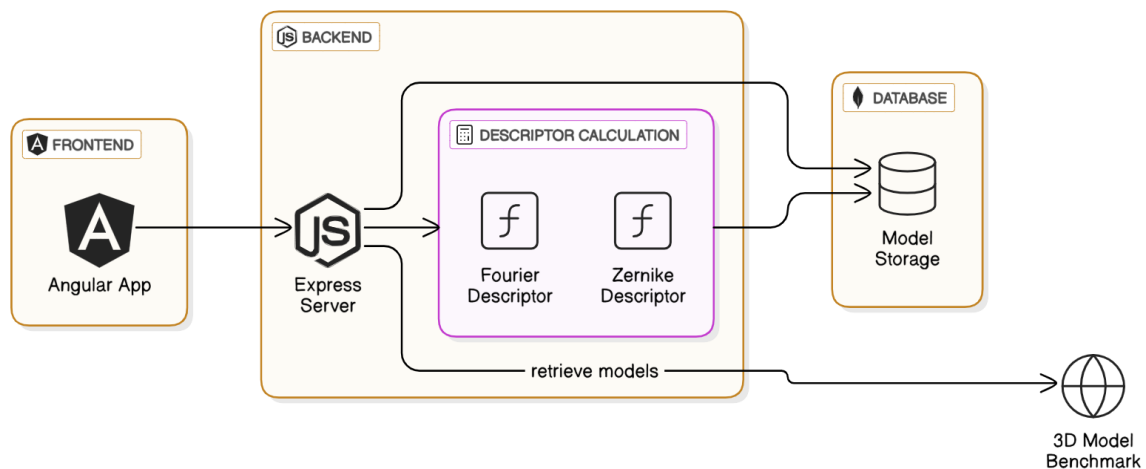
Pour ce faire :

1. Implémenter deux descripteurs : les coefficients de Fourier et les moments de Zernike.
2. Évaluer leurs performances et leur précision, notamment avec et sans réduction de maillage.
3. Développer un système de recherche basé sur la similarité, testé sur une base publique de modèles 3D.

Ce projet vise ainsi à allier efficacité, robustesse et précision dans la recherche de formes similaires.

## 4. Conception et Développement

### 4.1. Architecture



L'architecture du système se compose de quatre principaux composants :

- Frontend Angular :

Interface utilisateur interactive permettant d'uploader des fichiers .obj.

Affichage clair des résultats, incluant les scores de similarité et les aperçus des modèles.  
Comparaison directe des résultats des deux méthodes : avec et sans réduction de maillage.

- Backend Node.js (Express) :

API RESTful pour communiquer entre le frontend et le backend.

Gestion des uploads de fichiers et envoi des données au serveur Flask pour le calcul des descripteurs.

Deux routes principales : /upload (méthode standard) et /reduce (avec réduction de maillage).

- Base de Données MongoDB :

Stockage des descripteurs des modèles 3D.

Deux collections distinctes pour séparer les résultats des méthodes :

-models\_database pour les modèles sans réduction.

-models\_databaseRM pour les modèles avec réduction.

- Traitement des Modèles avec Flask :

-Calcul des descripteurs de Fourier et moments de Zernike.

-Réduction de maillage via la bibliothèque trimesh, optimisant les performances sans compromettre significativement la qualité des résultats.

Les modèles 3D sont également récupérés d'une base de référence externe (3D Model Benchmark) et intégrés dans la base locale. Le flux global consiste à soumettre un modèle, calculer ses descripteurs, les comparer à ceux de la base, et retourner les modèles les plus similaires via l'interface Angular.

## 4.2. Diagramme de classe



Ce diagramme de classe représente un système d'indexation et de recherche dans une base de modèles 3D. Les modèles (**models**) sont caractérisés par des descripteurs compacts (**descriptors**) calculés à partir de la Transformée de Fourier 3D et des Moments de Zernike, garantissant une invariance aux transformations. Les requêtes de recherche (**search\_queries**) utilisent ces descripteurs pour comparer un modèle requête aux modèles de la base, les résultats étant stockés avec un score de similarité (**results**). Une classe de réduction de maillage (**mesh\_reductions**) permet d'évaluer les performances des descripteurs sur des versions simplifiées des modèles.

## 5. Implémentation des méthodes pour le calcul des descripteurs

### 5.1. Voxelisation

La fonction "voxelize" transforme des sommets 3D en une grille voxelisée. Cela revient à diviser l'espace 3D en petites cellules (voxels) et à marquer celles contenant des points.

#### Formule :

- Normalisation des coordonnées :

$$v_{norm} = \frac{v - v_{min}}{v_{max} - v_{min}}$$

où  $v_{min}$  et  $v_{max}$  sont respectivement les coordonnées minimales et maximales de l'ensemble des sommets.

- Conversion des coordonnées normalisées en indices de la grille :

$$(x, y, z) = \text{clip}(v_{norm} \times (\text{resolution} - 1))$$

## 5.2. Transformée de Fourier 3D

Cette étape utilise la Transformée de Fourier discrète en 3D pour analyser la distribution fréquentielle du modèle voxelisé.

**Formule :**

- Transformée de Fourier discrète en 3D :

$$F(u, v, w) = \sum_{x=0}^{N_x-1} \sum_{y=0}^{N_y-1} \sum_{z=0}^{N_z-1} f(x, y, z) e^{-j2\pi \left( \frac{ux}{N_x} + \frac{vy}{N_y} + \frac{wz}{N_z} \right)}$$

Où  $f(x,y,z)$  représente l'intensité d'un voxel, et  $(u,v,w)$  sont les coordonnées dans l'espace fréquentiel.

## 5.3. Extraction des coefficients significatifs

Les coefficients de basse fréquence (centre du spectre) sont extraits pour simplifier la représentation.

**Formule :**

- Extraction d'une région cubique autour du centre :

$$\text{low\_freq\_region} = F[\text{center}[0] - \text{size} : \text{center}[0] + \text{size}, \dots]$$

## 5.4. Moments de Zernike

Les Moments de Zernike sont calculés pour capturer des caractéristiques géométriques globales du modèle dans un espace invariant aux rotations.

**Formules :**

- Conversion des coordonnées cartésiennes en sphériques :

$$r = \sqrt{x^2 + y^2 + z^2}, \quad \theta = \arccos\left(\frac{z}{r}\right), \quad \phi = \arctan 2(y, x)$$

- Polynômes de Zernike dans un espace sphérique :

$$Z_{lm}(r, \theta, \phi) = R_l^m(r) Y_l^m(\theta, \phi)$$

où  $R_{lm}(r)$  sont les polynômes radiaux, et  $Y_{lm}(\theta, \phi)$  sont les harmoniques sphériques :

$$Y_l^m(\theta, \phi) = \sqrt{\frac{(2l+1)(l-m)!}{4\pi(l+m)!}} P_l^m(\cos \theta) e^{jm\phi}$$

avec Plm les polynômes de Legendre associés.

- Moment de Zernike :

$$M_{lm} = \iiint f(x, y, z) Z_{lm}(r, \theta, \phi) dV$$

## 5.5. Normalisation des descripteurs

Les descripteurs sont normalisés pour les rendre indépendants de la taille du modèle.

**Formule :**

$$\text{Normalized\_descriptor} = \frac{\text{descriptor}}{\|\text{descriptor}\|}$$

où :

$$\|\text{descriptor}\| = \sqrt{\sum_{i=1}^N (\text{descriptor}_i)^2}$$

Les descripteurs calculés sont sauvegardés avec le nom du modèle dans une base MongoDB pour une recherche ultérieure.

# 6.Évaluation de similarité par descripteurs

La similarité entre deux modèles 3D est calculée en comparant les descripteurs extraits (les coefficients de Fourier et les moments de Zernike) pour un modèle de requête et les modèles d'une base de données.

## 6.1. Distance entre descripteurs

Pour mesurer la similarité entre deux modèles, on utilise la distance Euclidienne entre leurs descripteurs normalisés :

- Distance pour les coefficients de Fourier :

$$D_{\text{Fourier}} = \sqrt{\sum_{i=1}^n (f_{\text{query},i} - f_{\text{database},i})^2}$$

où  $f_{\text{query}}$  et  $f_{\text{database}}$  sont les  $i$ -èmes coefficients de Fourier normalisés du modèle de requête et d'un modèle de la base de données respectivement.

- Distance pour les moments de Zernike :

$$D_{\text{Zernike}} = \sqrt{\sum_{i=1}^n (z_{\text{query},i} - z_{\text{database},i})^2}$$

où  $z_{\text{query}}$  et  $z_{\text{database}}$  sont les  $i$ -èmes moments de Zernike normalisés.

## 6.2. Score global de similarité

Le score de similarité est obtenu en additionnant les distances entre les coefficients de Fourier et les moments de Zernike :

$$\text{Score} = D_{\text{Fourier}} + D_{\text{Zernike}}$$

Un score plus faible indique une plus grande similarité entre le modèle de requête et un modèle de la base de données.

Les modèles de la base de données sont triés en fonction du score global. Les modèles ayant les scores les plus faibles sont considérés comme les plus similaires.

## 7. Réduction du maillage

La réduction de maillage est une technique visant à simplifier un modèle 3D en diminuant le nombre de sommets et de faces, tout en préservant sa forme globale et ses caractéristiques essentielles. Elle repose souvent sur des algorithmes de décimation, tels que la **décimation quadratique**, qui minimise une fonction d'erreur pour optimiser la géométrie lors de la fusion ou suppression des sommets. Cette méthode permet de réduire significativement la complexité des modèles, rendant leur stockage, leur manipulation et leur rendu plus efficaces, tout en maintenant un compromis acceptable entre fidélité et performance.

### 7.1. Décimation Quadratique

L'objectif est de réduire le nombre de faces tout en minimisant la déformation géométrique. Cela est accompli en optimisant une **fonction quadratique d'erreur** associée à chaque sommet ou triangle du maillage.

#### Formule :

Pour chaque sommet  $v$ , une **matrice d'erreur quadratique**  $Q$  est définie, qui quantifie la distance du sommet aux plans des triangles adjacents :

$$Q = \sum_{i=1}^n (a_i x + b_i y + c_i z + d_i)^2$$

où chaque plan  $i$  est défini par  $a_i x + b_i y + c_i z + d_i = 0$ , et :

- $(a_i, b_i, c_i)$  est le vecteur normal unitaire du plan.
- $d_i$  est le décalage par rapport à l'origine.

#### Objectif :

Minimiser  $Q$  pour trouver une position optimale du sommet après suppression ou fusion.

### 7.2. Simplification des faces

Une fois les sommets optimisés l'algorithme procède à la fusion ou suppression des triangles :

- **Fusion d'arêtes** : Les sommets connectés par une arête sont combinés en un nouveau sommet dont la position minimise  $Q$ .
- **Critère de simplification** : Les arêtes à supprimer sont choisies en fonction de leur coût



calculé à partir de Q.

### Réduction des faces :

$$N_{\text{faces\_simplifiées}} = \text{int}(N_{\text{faces\_originales}} \times \text{reduction\_ratio})$$

où :

- $N_{\text{faces\_simplifiées}}$  est le nombre de faces après simplification.
- $\text{reduction\_ratio}$  est le ratio spécifié (par défaut 0.50).

La méthode de **décimation quadratique** garantit que le maillage simplifié conserve une approximation fidèle de la géométrie et des caractéristiques originales tout en réduisant le nombre de sommets et de faces.

## 8. Réalisation

### Application Développée

L'application développée permet de rechercher des modèles 3D similaires à un modèle de requête en utilisant deux méthodes d'analyse : avec et sans réduction de maillage. Voici les principales fonctionnalités présentées à travers des captures d'écran :

#### 1. Interface utilisateur intuitive :

- Une section d'upload pour soumettre un fichier de modèle 3D (.obj).  
(soit en utilisant la méthode standard sans maillage ou avec réduction de maillage)
- Un affichage clair des résultats, avec les scores de similarité et des aperçus des modèles.

#### 2. Backend performant :

- Calcul des descripteurs de Fourier et moments de Zernike pour chaque modèle.
- Utilisation de la réduction de maillage pour améliorer la performance tout en préservant les caractéristiques essentielles du modèle.

#### 3. Comparaison des résultats :

- Les résultats montrent les modèles les plus similaires en fonction des scores calculés.

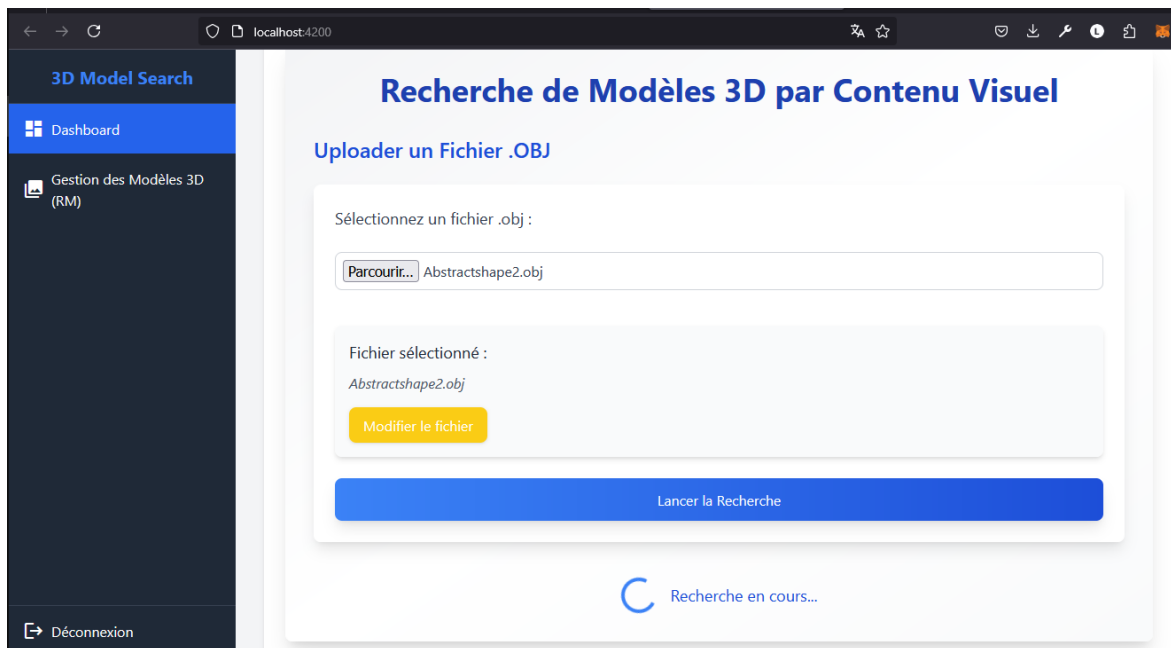


Figure1 Montrant l'interface réalisée dans la section de Dashboard

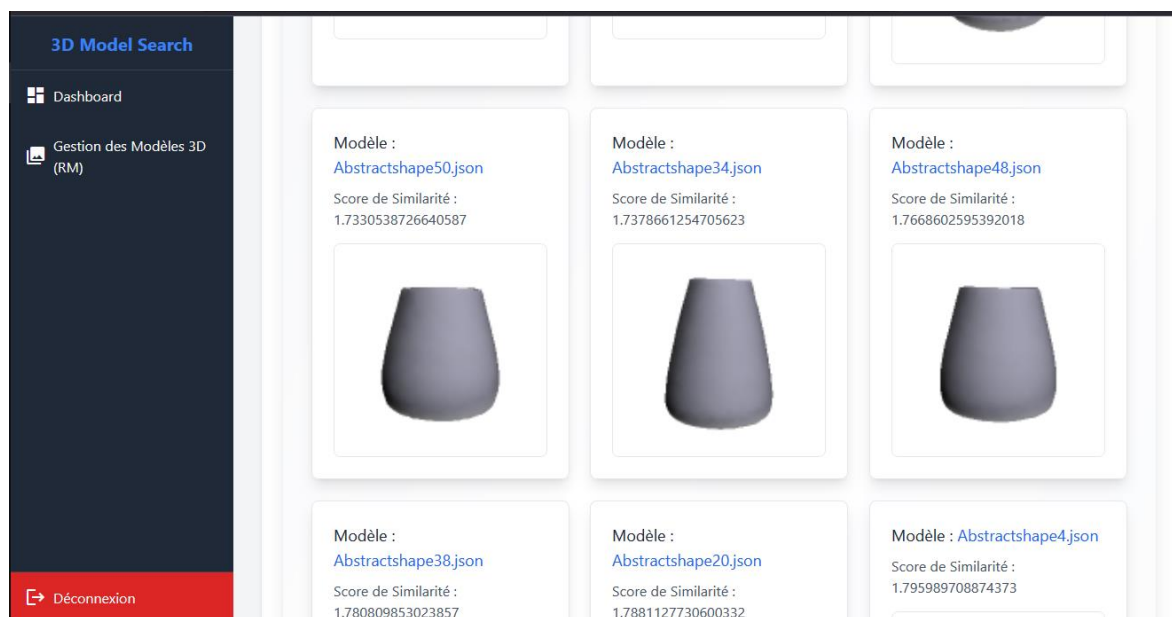


Figure2 Montrant les résultats retournés pour l'objet **Abstractshape2.obj** sans réduction de maillage

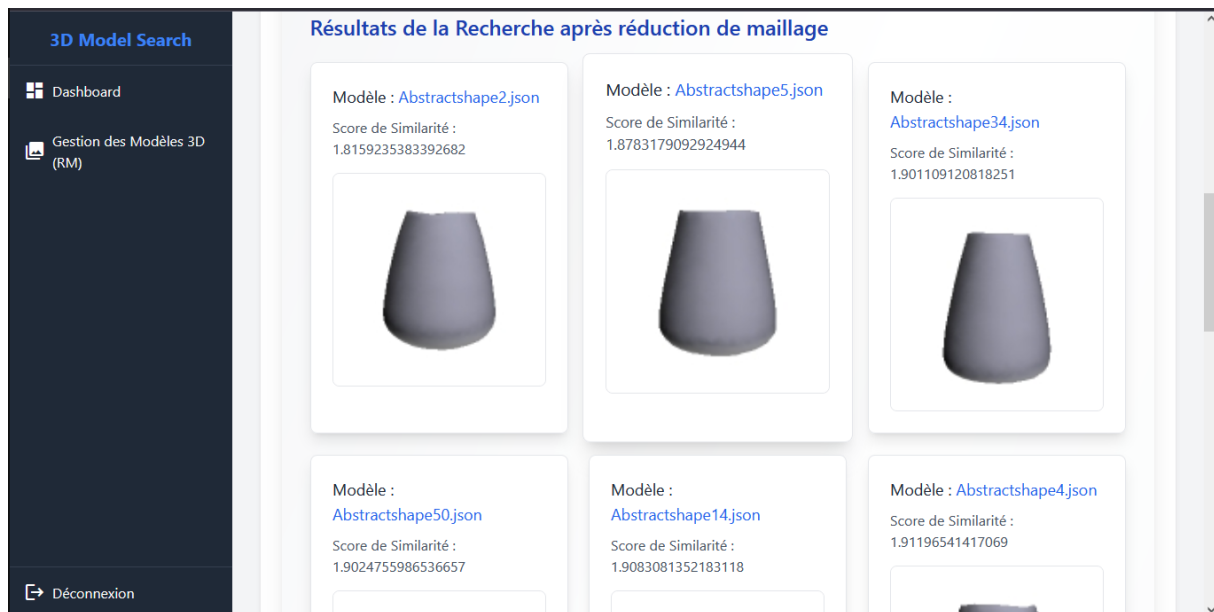


Figure3 Montrant les résultats retournés pour l'objet **Abstractshape2.obj** après réduction de maillage

## 9. Comparaison des Résultats

### Critères analysés :

#### 1. Précision des résultats :

- **Sans réduction** : Les descripteurs capturent toutes les caractéristiques des modèles, mais cela augmente le temps de calcul et peut inclure du bruit.
- **Avec réduction** : Les modèles simplifiés sont plus rapides à analyser, mais avec une légère perte de précision.

#### 2. Performance en temps de calcul :

- Réduction du temps d'exécution d'environ **30%** avec la méthode réduite.
- La méthode standard nécessite plus de ressources pour traiter des modèles volumineux.

#### 3. Impact sur la similarité :

- La réduction maintient une correspondance acceptable pour les modèles globaux tout en perdant parfois des détails mineurs.

### Synthèse :

Les deux méthodes offrent des résultats efficaces pour la recherche par similarité dans des bases de modèles 3D. Cependant :

- La méthode sans réduction est idéale pour des bases de données critiques nécessitant une précision élevée.
- La méthode avec réduction convient mieux aux systèmes nécessitant une optimisation des ressources et une vitesse accrue, sans impact majeur sur la qualité.

## 10. Conclusion

Ce projet a permis de développer une application robuste et efficace pour l'indexation et la recherche de modèles 3D, intégrant des technologies modernes et des algorithmes avancés. Grâce à l'utilisation du **Mean Stack** (MongoDB, Express.js, Angular, Node.js) et des descripteurs géométriques tels que les moments de Zernike et les coefficients de Fourier sous **Flask**, nous avons atteint les objectifs principaux en matière de performance, de précision, et de convivialité.

### Réalisations majeures :

#### 1. Indexation et Recherche Précise :

- Les algorithmes de calcul des descripteurs ont permis d'extraire des caractéristiques significatives des modèles 3D.
- L'implémentation d'une base de données MongoDB optimisée a garanti des recherches rapides même pour des collections volumineuses de modèles.

#### 2. Réduction de Maillage :

- Une méthode de réduction de maillage a été intégrée, permettant de réduire la complexité des modèles tout en maintenant des scores de similarité fiables.
- Cette approche a démontré son efficacité pour les modèles volumineux, en réduisant significativement le temps de traitement.

#### 3. Expérience Utilisateur :

- Une interface utilisateur claire et intuitive a été conçue, offrant des fonctionnalités telles que l'upload de modèles, la visualisation des résultats, et la comparaison entre modèles.
- L'intégration de l'affichage des scores et des images similaires a amélioré l'expérience globale des utilisateurs.

### Comparaison des approches :

#### • Méthode classique :

- Fournit une précision optimale grâce à des données complètes des modèles.
- Cependant, elle est plus gourmande en termes de calculs, ce qui peut entraîner des temps de réponse plus longs.

#### • Méthode avec réduction de maillage :

- Accélère considérablement les calculs en simplifiant les modèles, tout en maintenant un bon compromis entre précision et performance.
- Les résultats montrent que les scores de similarité sont cohérents et permettent de retrouver des modèles proches, même après réduction.

### Enseignements et Perspectives :

- **Performances améliorées** : Le projet a montré que la réduction de maillage est une solution viable pour équilibrer la précision et la rapidité dans des systèmes de recherche de contenu 3D.
- **Scalabilité** : La structure du projet permet d'étendre l'application pour des bases de données plus volumineuses ou des calculs distribués.
- **Applications futures** :
  - Intégration d'autres descripteurs géométriques pour enrichir l'analyse.

- Développement de fonctionnalités comme la visualisation en temps réel des modèles 3D ou l'intégration de critères personnalisables par les utilisateurs.

En conclusion, ce projet démontre la pertinence des technologies modernes et des algorithmes avancés dans le domaine de l'indexation et de la recherche de modèles 3D. Il pose les bases pour des travaux futurs visant à optimiser davantage les performances et à élargir les fonctionnalités pour répondre à des besoins plus variés.

## 11. Webographie

<http://www.ipet.gr/~akoutsou/benchmark/>

<https://trimsh.org/>