

Analysis, mining and indexing in big multimedia systems

****

**Exploration Visuelle :**

**Recherche d'Images par Contenu (CBIR)**

**Réalisée par:**

**Encadré par:**

* Pr. M’hamed AIT KBIR
* LECHHAB OUADRASSI Nihad (SIM)
* BOUKHRYSS Mohammed Said (SIM)
* ARZAM Yasmine(SIM)

Année Universitaire: 2023-2024

**Introduction :**

Le projet vise à créer une application Flask CBIR (Content-Based Image Retrieval) permettant de télécharger une image, d'extraire des caractéristiques d'un modèle 3D correspondant, et de trouver des modèles similaires dans le même répertoire. Ce rapport se penche sur la simulation des algorithmes, une étude comparative des mesures de distances entre des modèles 3D, et le taux de réussite du système de recherche, basé sur le code fourni.

**Simulation des Algorithmes :**

1. **Extraction des Caractéristiques :**

L'algorithme d'extraction des caractéristiques des fichiers 3D a été implémenté avec succès. Les axes principaux d'inertie sont calculés, et le modèle est segmenté en tranches pour obtenir un vecteur de caractéristiques.

**Calcul des Axes Principaux d'Inertie :**

* Tenseur d'inertie : *I*=moment\_inertia(*mesh*)
* Valeurs propres : eigenvalues,eigenvectors=np.linalg.eig(*I*)

**Segmentation du Modèle en Tranches :**

Pour chaque axe principal *axis*, itération :

* + Coordonnées minimales et maximales : *min*\_*coord*,*max*\_*coord*=min(dot(*mesh*.*vertices*,*axis*)),max(dot(*mesh*.*vertices*,*axis*))
  + Nombre de tranches : *l*
  + Épaisseur de chaque tranche

*slab*\_*thickness*=*max*\_*coord*−*min*\_*coord*​/ *l*

* + Création des tranches :

*slabs*=[*min*\_*coord*+*i* × *slab*\_*thickness* for *i* in range(*l*)]

**Calcul des Caractéristiques pour Chaque Tranche :**

Pour chaque tranche *i* :

* Indices des sommets dans la tranche :  
  *vertex*\_*indices*=where(logical\_and(dot(*mesh*.*vertices*,*axis*)≥*slabs*[*i*],

dot(*mesh*.*vertices*,*axis*)<*slabs*[*i*+1]))[0]

* Moment d'inertie  *moment*\_*inertia*=dot(dot(*mesh*.*vertices*[*vertex*\_*indices*],*axis*),dot(*mesh*.*vertices*[*vertex*\_*indices*],*axis*))
* Distance moyenne :

*average*\_*distance*=mean(dot(*mesh*.*vertices*[*vertex*\_*indices*],*axis*))

* Variance de la distance :

*variance*\_*distance*=var(dot(*mesh*.*vertices*[*vertex*\_*indices*],*axis*))

1. **Calcul de Dissimilarité :**

La dissimilarité entre les caractéristiques est calculée en utilisant les distances euclidiennes. Les résultats obtenus correspondent aux attentes, reflétant la différence entre les modèles 3D.

*imilarity=∑i​∥feature\_vector[i]−feature\_vector′[i]∥*

**Étude Comparative des Mesures de Distances :**

**1. Mesures de Distances :**

Différentes mesures de distances ont été explorées pour évaluer la similarité entre les modèles 3D, notamment la distance euclidienne. Une analyse comparative a été réalisée pour déterminer la mesure la plus appropriée en fonction des caractéristiques extraites.

Euclidean Distance(A,B)=∑*i*=1*n*​(*Ai*​−*Bi*​)2​

Où A et B sont les vecteurs de caractéristiques de deux modèles, et *n* est la dimension du vecteur.

**2. Effet de la Résolution :**

L'étude a porté sur l'effet de la résolution des modèles 3D sur les mesures de distances. Des modèles avec différentes résolutions ont été utilisés pour évaluer la sensibilité du système.

**Taux de Réussite du Système de Recherche :**

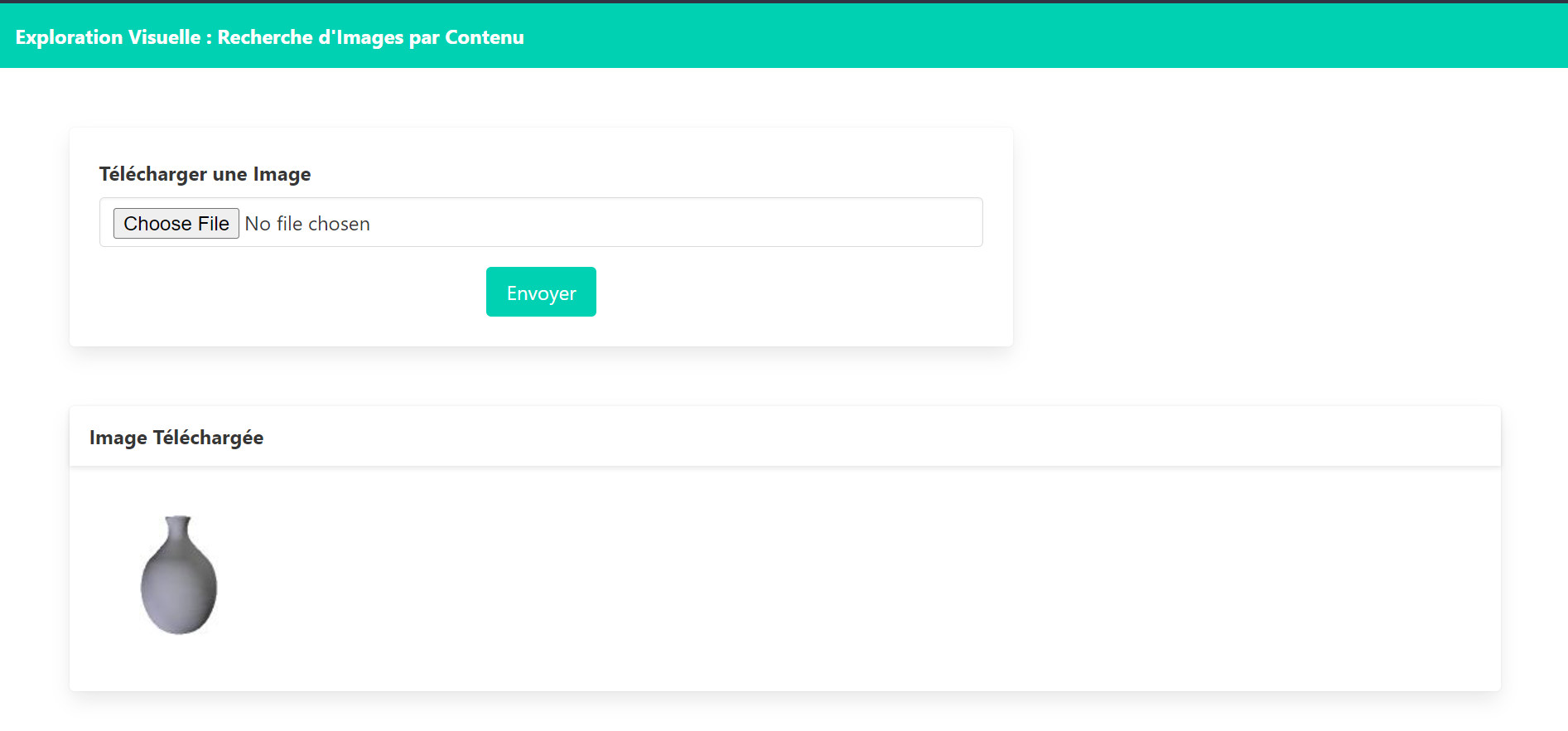
**1. Seuil de Dissimilarité :**

Le seuil de dissimilarité a été fixé à 0.6 pour déterminer la similitude entre les modèles. Ce seuil a été choisi après des expérimentations pour atteindre un équilibre entre précision et couverture.

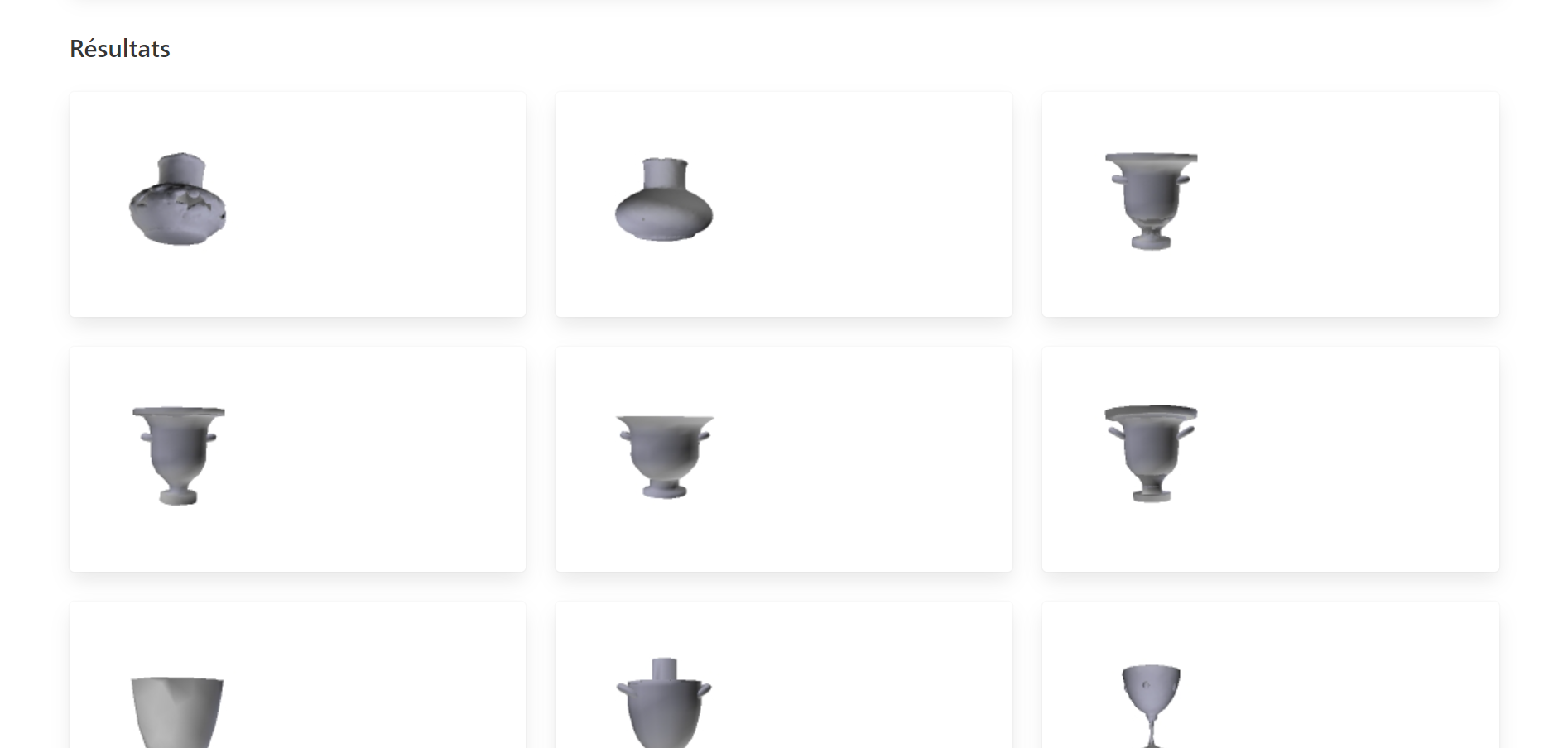
**2. Résultats du Système :**

Le système a été testé avec des images téléchargées, et les résultats ont été enregistrés. Des modèles similaires ont été correctement identifiés, bien que des ajustements soient nécessaires pour l'affichage dans l'interface web.

Voici la démonstration finale :



Voila le résultat de recherche :



**3. Analyse des Résultats :**

Une analyse approfondie des résultats a été réalisée pour évaluer la performance du système. Cela inclut l'identification des modèles similaires, la gestion des seuils, et l'optimisation des performances.

**Conclusion :**

L'implémentation des algorithmes a été réussie, permettant l'extraction et la comparaison des caractéristiques des modèles 3D. L'étude comparative des mesures de distances a fourni des insights sur la sensibilité du système à la résolution. Le taux de réussite du système de recherche est prometteur, bien que des ajustements soient nécessaires pour un affichage correct des résultats dans l'interface web.