

Mohamed Nefsi

300305042

Similarity image search

OOP.java

Introduction

Le programme Java développé pour la recherche d'images similaires démontre efficacement l'application du paradigme orienté objet (POO) pour résoudre des problèmes complexes. En structurant la solution autour de classes distinctes, chacune représentant un concept ou une fonctionnalité spécifique du domaine de la recherche d'images, il illustre comment la POO facilite l'organisation du code, la réutilisation des composants et l'extension du système. Les classes pour le traitement d'images, le calcul des caractéristiques visuelles et la comparaison de similarité travaillent ensemble de manière cohérente, montrant la force du paradigme orienté objet dans la conception de solutions modulaires et maintenables pour des défis informatiques avancés.

Classes utilise

ColorHistogram.java

Cette classe est responsable de la création et de la manipulation d'histogrammes de couleurs pour les images. Les histogrammes de couleurs sont une représentation cruciale dans le domaine du traitement d'images, utilisée pour quantifier la distribution des couleurs dans une image. La classe ColorHistogram :

- Calcule et stocke les valeurs normalisées de l'histogramme.
- Permet une représentation concise des caractéristiques de couleur d'une image, facilitant la comparaison des images basée sur leur contenu en couleur.

ColorImage.java

ColorImage sert à charger et à représenter des images en couleur, en fournissant un accès direct aux valeurs des pixels et à diverses propriétés de l'image telles que la largeur, la hauteur et la profondeur de couleur. Cette classe est essentielle pour le traitement initial des images avant d'effectuer des comparaisons ou des analyses plus approfondies.

ImageDistance.java

La classe ImageDistance détermine et stocke la distance ou la similarité entre deux images. Cette mesure de similarité est fondamentale pour la recherche d'images similaires, permettant de classer les images en fonction de leur ressemblance avec une image de référence. En implémentant l'interface Comparable, ImageDistance facilite le tri des images par similarité.

SimilaritySearch.java

Le cœur du programme, SimilaritySearch, orchestre la recherche de similarité entre une image de requête et un ensemble d'images candidates. En utilisant les fonctionnalités fournies par les autres classes, cette classe effectue les étapes suivantes :






- Chargement et traitement des images de requête et des images candidates.
- Calcul des mesures de similarité entre l'image de requête et chaque image candidate.
- Tri et sélection des images les plus similaires pour la présentation des résultats.

Exemple des resultats

1. Les 5 premiers images similaires à q00.jpg:

Command line : java SimilaritySearch q00.jpg ImageDataset2_15_20





```
• les meilleurs5 similar a sont queryImages/q00.jp
2462.jpg.txt - Similarite: 0.6491030092592595
3756.jpg.txt - Similarite: 0.6620254629629632
3714.jpg.txt - Similarite: 0.668431712962963
3806.jpg.txt - Similarite: 0.7074537037037038
o 1144.jpg.txt - Similarite: 1.0
```

				
2462	3756	3714	3806	1144

2. Les 5 premiers images similaires à q01.jpg:

Command line : java SimilaritySearch q01.jpg ImageDataset2_15_20

```
les meilleurs5 similaire a sont queryImages/q01.jpg:
2592.jpg.txt - Similarite: 0.5049710648148149
1875.jpg.txt - Similarite: 0.5282986111111114
3553.jpg.txt - Similarite: 0.5365624999999999
2536.jpg.txt - Similarite: 0.561255787037037
3588.jpg.txt - Similarite: 0.5703645833333333
```

				
2592	1875	3553	2536	3588