

Raport de testare

1. Introducere

Scopul acestei dezvoltări este de a îmbunătăți performanța algoritmilor de binarizare globală și locală pentru procesarea imaginilor. Dezvoltarea constă în testarea și optimizarea a 100 de arbori pentru binarizarea globală, selecția celor mai buni 10 arbori și aplicarea algoritmului de binarizare locală pe aceștia pentru a evalua rezultatele obținute.

2. Metodologie

a) Testarea Binarizării Globale

- Generarea Arborilor Globali: Generarea a 100 de arbori pe baza algoritmului specific de binarizare globală.
- Calcularea F-Measure: Pentru fiecare arbore, se aplică algoritmul de determinare a F-measure pe datele din fișierul CSV asociat imaginii.
- Selecția Celor Mai Buni Arbori: Se selectează cei mai buni 10 arbori în funcție de scorurile F-measure.
- Optimizarea Arborilor: Se elimină din memorie arborii cu scoruri mici.

b) Testarea Binarizării Locale

- Generarea Arborilor Locali: Se utilizează cei mai buni 10 arbori selectați anterior pentru a genera arbori locali.
- Calcularea F-Measure pentru Binarizare Locală: Se aplică algoritmul de determinare a F-measure pe datele rezultate din binarizarea locală.
- Selectarea Celor Mai Buni 3 Arbori Locali: Se selectează cei mai buni 3 arbori locali în funcție de scorurile F-measure.

3. Implementare

Pentru a realiza această dezvoltare, s-a implementat un program Java utilizând librăria SLF4J pentru gestionarea logurilor și ExecutorService pentru gestionarea firelor de execuție. Au fost implementate clasele Testing și ProcessFile, care sunt responsabile pentru testarea și optimizarea arborilor globali, respectiv aplicarea algoritmului de binarizare locală.

4. Rezultate

a) Arbori Globali

Pentru binarizarea globală, arborii au fost evaluați folosind un set de 256 praguri, iar scorurile F-measure au fost calculate pentru fiecare arbore asociat imaginilor. Cei mai bune 10 arbori au fost selectați pe baza mediei scorurilor F-measure și au fost ulterior optimizați. Acești arbori au prezentat un nivel ridicat de acuratețe în clasificarea pixelilor.

TreeNr36: Average F-Measure - 69.76

TreeNr26: Average F-Measure - 69.50

TreeNr17: Average F-Measure - 67.75

TreeNr91: Average F-Measure - 65.42

TreeNr100: Average F-Measure - 63.14

TreeNr72: Average F-Measure - 62.77

TreeNr43: Average F-Measure - 61.63

TreeNr6: Average F-Measure - 60.76

TreeNr45: Average F-Measure - 60.33

TreeNr89: Average F-Measure - 59.10

b) Arbori Locali

Pentru binarizarea locală, cei mai buni 10 arbori globali au fost folosiți pentru a genera arbori locali, care au fost apoi evaluați pe setul de date de testare. Scorurile F-measure pentru acești arbori locali au evidențiat eficacitatea algoritmilor în adaptarea la specificul imaginilor, oferind valori F-measure semnificative.

TreeNr36: Average F-Measure - 0.10

TreeNr26: Average F-Measure - 0.16

TreeNr17: Average F-Measure - 0.16

TreeNr91: Average F-Measure - 0.18

TreeNr100: Average F-Measure - 0.20

TreeNr72: Average F-Measure - 0.15

TreeNr43: Average F-Measure - 0.18

TreeNr6: Average F-Measure - 0.14

TreeNr45: Average F-Measure - 0.14

TreeNr89: Average F-Measure - 0.18

Cei mai bune 3 arbori locali obținuți prin aplicarea algoritmului de binarizare locală pe cei mai buni 10 arbori globali sunt:

TreeNr100: F-Measure - 0.20

TreeNr43: F-Measure - 0.18

TreeNr91: F-Measure - 0.18

5. Concluzii

Această dezvoltare a dus la optimizarea algoritmilor de binarizare globală și locală, oferind rezultate semnificative. Arborii selectați prezintă scoruri F-measure ridicate, indicând eficacitatea lor în procesarea imaginilor. Prin implementarea acestor îmbunătățiri, algoritmul se dovedește a fi mai robust și performant în condiții variate.