Univestitatea POLITEHNICA Bucuresti

Facultatea de Automatica si Calculatoare

Specializarea Calculatoare si Tehnologia Informatiei



**Voting-Based Image Binarization**

**README**

(Proiect Managementul Proiectelor Software)

**Indrumator:**

Saca Octavian

**Componenta echipei:**

Andreea Dragomir 341 C5

Mihai Trifu 341 C5

Dragos Godeanu 341 C5

**Bucuresti**

**2013**

Cuprins:

[1. Rolul Documentului 3](#_Toc375139678)

[2. Rolurile in echipa 3](#_Toc375139679)

[3. Despre proiect 3](#_Toc375139680)

[4. Algoritm 3](#_Toc375139681)

[5. Mod de rulare 4](#_Toc375139682)

[6. Observatii 4](#_Toc375139683)

# Rolul Documentului

Acest document are rolul de a explica

* modul in care a fost organizata echipa
* algoritmul folosit
* modul de rulare al programului

# Rolurile in echipa

Echipa a fost alcatuita din 3 membrii, fiecare avand cate un rol in cadrul dezvoltarii proiectului:

* **Andreea Dragomir (Tester)** – s-a ocupat de realizarea SDD-ului, Gantt si WBS, a cautat imagini pentru testare si s-a ocupat de testarea propriu-zisa
* **Mihai Trifu (Developer)** – s-a ocupat de alegerea unuia dintre algoritmii propusi de PM si de implementarea acestuia
* **Dragos Godeanu (Project manager) –** s-a ocupat de organizarea timpului, trasarea sarcinilor, crearea documentelor “activity report”, code review, research in ceea ce priveste algoritmul folosit

Desi teoretic fiecare a avut sarcini bine definite, fiecare la randul sau a fost ajutat la realizarea acestora de ceilalti doi membri.

# Despre proiect

Scopul aplicatiei este de a obtine forma binarizata a unor imagini cu text deteriorat, avand o separare cat mai clara intre foreground si background, astfel incat pe rezultat sa se poata aplica algoritmi OCR.

# Algoritm

Pentru a obtine imaginea binarizata, am folosit algoritmul Bernsen. Modul de functionare al acestuia este urmatorul:

* Imaginii initiale i se face padding de jur imprejur cu oglinditul ei raportat la latura respective
* Imaginea este impartita in mai multe imagini mai mici, cu latura de dimensiunea paddingului
* In fiecare dintre aceste imagini
  + - se calculeaza care exte pixelul cu valoarea maxima si minima
    - se calculeaza diferenta si media acestora
    - se verifica daca diferenta este mai mare decat un anumit thresholdconstant setat initial
      * + daca da, thresholdul va fi media dintre gri min si gri max
        + daca nu, thresholdul va fi o constanta setata initial
      * daca valoarea pixelului current este mai mare decat thresholdul, pixelul binarizat va fi negru, iar daca este mai mica, alb

Matricea de confidenta a fost calculata pentru fiecare imagine mica, dupa urmatoarea formula:

unde midGray este media dintre minGray si maxGray, ce reprezinta minimul, respectiv maximul dintre valorile pixelilor din acea imagine.

# Mod de rulare

Pentru a rula programul, executabilul “BAM3.exe”, ce poate fi gasit in folderul Exe\_file alaturi de libraria FreeImage.dll, va fi apelat din command prompt cu 3 parametrii:

* numele imaginii de intrare
* numele imaginii binarizare rezulata
* numele imaginii de confindenta rezulata

# Observatii

Nu stiu din ce motive, dar imaginile rezultate difera uneori atunci cand codul este rulat din Visual Studio fata de atunci cand este rulat direct executabilul din cmd. Diferenta apare doar pe ultima coloana, respective ultimul rand de mini-imagini procesate in cadrul celei originale. Aceasta diferenta se poate observa cel mai bine intre imaginile Ex1\_VS.TIF si Ex1\_cmd.TIF din folderul output, unde numarul de mini-imagini in care se imaparte cea originala este mic

Tot in folderul output se gasesc diferite outputuri rezultate in functie de cum au fost setati acei parametrii (dimensiunea paddingului – si implicit a mini-imaginilor, thresholdul constant si constanta de gri). Desi teoretic binarizarea ar functiona mai bine cand paddingul este mare, acest lucru nu se intampla decat pentru imaginile care au o oarecare uniformitate ( a se vedea Ex1\_30), insa pentru alte imagini, raman multe zone cu zgomot (a se vedea Ex4\_30) . Daca thresholdul setat manual are o valoare mai mare, pentru a incerca eliminarea acelor zone, acest lucru va dauna in cazul primului exemplu, deoarece imaginea va fi prea stearsa (a se vedea Ex4\_128 respectiv Ex1\_128). In concluzie, am preferat sa folosesc un padding mai mic astfel incat binarizarea sa functioneze pe ambele tipuri de imagini, cu rezultate medii.