# DOCUMENTUL DE PROIECTARE A SOLUTIEI APLICATIEI SOFTWARE

# Contents

# 1. Introducere

## 1.1 Scopul Proiectului

Documentul prezent are ca scop descrierea precisa si completa a functionalitatii unei aplicatii software de binarizare a imaginilor. Prin aceasta aplicatie se intelege un sistem ce ofera o varianta cat mai clara a unui document vechi, convertit in format electronic.

Acest document se vrea a fi inteles si utilizat atat de echipa de dezvoltatori cat si de utilizatorii aplicatiei.

## 1.2 Definirea termenilor tehnici

*Digitalizare* - actiunea prin care un document trece dintr-un format fizic intr-unul electronic.

binarizare - actiunea prin care se transpune o imagine intr-un format ce contine doar pixeli albi si negri.

*Pixel* - cel mai mic element (punctual) în care se poate descompune o imagine.

*BAM* - Binarization Algorithm Module

*VBAM* - Voting Binarization Algorithm Module

*White-box testing*: constă în testarea şi verificarea codului propiu-zis; în timpul implementării fiecărei clase developer-ul care se ocupă de clasa respectivă va face teste pentru a se asigura că fiecare metodă returnează rezultatul corect.

*Black-box testing*: reprezintă majoritatea procesului de testare; aceasta se efectuează după ce toate componentele proiectului sunt realizate şi constă în introducerea input-ului şi verificarea output-ului, daca este cel corespunzator.

# 2. Obiective

## 2.1 Situatie actuala

In decursul ultimelor decenii, crearea si depozitarea documentelor a trecut de la formatul fizic la cel electronic. Acest fapt a schimbat radical modul in care oamenii interactioneaza cu datele. Desi documentele create recent beneficiaza atat de suport fizic cat si electronic, cele din trecut se regasesc doar in forma lor fizica. Acest lucru poate fi un dezavantaj intrucat exemple de astfel de documente pot fi manuscrise sau tiparituri ce contin informatii importante ce s-ar dori sa fie procesate in mod automat.

## 2.2 Scopul aplicatiei

Aplicatia doreste sa permita digitalizarea documentelor ce nu dispun de o forma electronica printr-un algoritm de binarizare ce va fi indeajuns de complex incat sa ofere un output cat mai lizibil indiferent de gradul de degradare al documentului original.

## 2.3 Contextul aplicatiei

*Binarizarea* unui document este un proces complex care presupune mai multi pasi de procesare. Aceasta pleaca de la o fotografie alb-negru a documentului pentru a beneficia de o diferentiere cat mai clara intre fundal si continut.

Datorita diferentelor semnificative dintre documentele ce se vor a fi digitalizate, nu s-a obtinut inca niciun algoritm ce poate garanta o binarizare corecta pentru orice manuscris.

# 3. Continutul documentului

In prima parte este descris sumar proiectul si scopul acestuia, precum si contextul actual al aplicatiei.

In cea de-a doua parte se prezinta modelul datelor folosit in aplicatie. Acesta consta in structurile, formatul fisierelor, etc. utilizate in dezvoltarea programului.

In a treia parte este descris modelul arhitectural și modelul componentelor. Acesta reprezinta sabloane arhitecturale folosite, diagrama de arhitectură, descrierea componentelor, restricțiile de implementare si interacțiunea dintre componente.

In a patra parte este descris modul in care utilizatorul interactioneaza cu aplicatia.

In ultima parte se prezinta modalitatile prin care aplicatia va fi testata.

# 4. Modelul datelor

## 4.1 Structuri de date globale

Instanta clasei Manager, care porneste executabilele modulelor BAM si contine timpul si resursele consumate de fiecare in parte, pe care le va folosi in stabilirea algoritmului castigator. Totodata, contine si o instanta a analyzer-ului, care va implementa sistemul de votare. Analyzer-ul va retine imaginile initiale intr-o structura de tip vector.

## 4.2 Structuri de date de legatura

Legatura intre utilizator si sub-modulul Manager, componenta a VBAM, se realizeaza prin intermediul parametrului dat in linia de comanda, care reprezinta calea catre imaginea input.

Legatura intre sub-modulul Manager transmite modulelor BAM calea catre imaginea input, dupa care preia timpii, resursele consumate si rezultatele pe care le transmite Analyzer-ului.

## 4.3 Formatul fisierelor utilizate

Imaginile vor avea format de tip .PGM.

Pentru mai multe detalii despre formatul detaliat al acestui tip de fisiere puteti gasi la adresa:

http://users.wpi.edu/~cfurlong/me-593n/pgmimage.html

Structura unui astfel de fisier va fi urmatoarea:

P2/P4

N M

[matrice valori pixeli]

Imaginile input si matricile de factori vor avea tipul P2, pentru ca vor contine 8biti/pixel.

Imaginile binarizate (atat cele intoarse de modulele BAM, cat si imaginea rezultat), va avea tipul P4, pentru ca vor avea 1bit/pixel.

N reprezinta numarul de linii si M reprezinta numarul de coloane ale matricei.

[matrice valori pixel] = matricea cu toate valorile pixelilor (ori 0/1 – pentru formatul P4, ori valori cuprinse intre 0-255 – pentru formatul P2).

# 5. Modelul architectural si modelul componentelor

## 5.1 Arhitectura sistemului

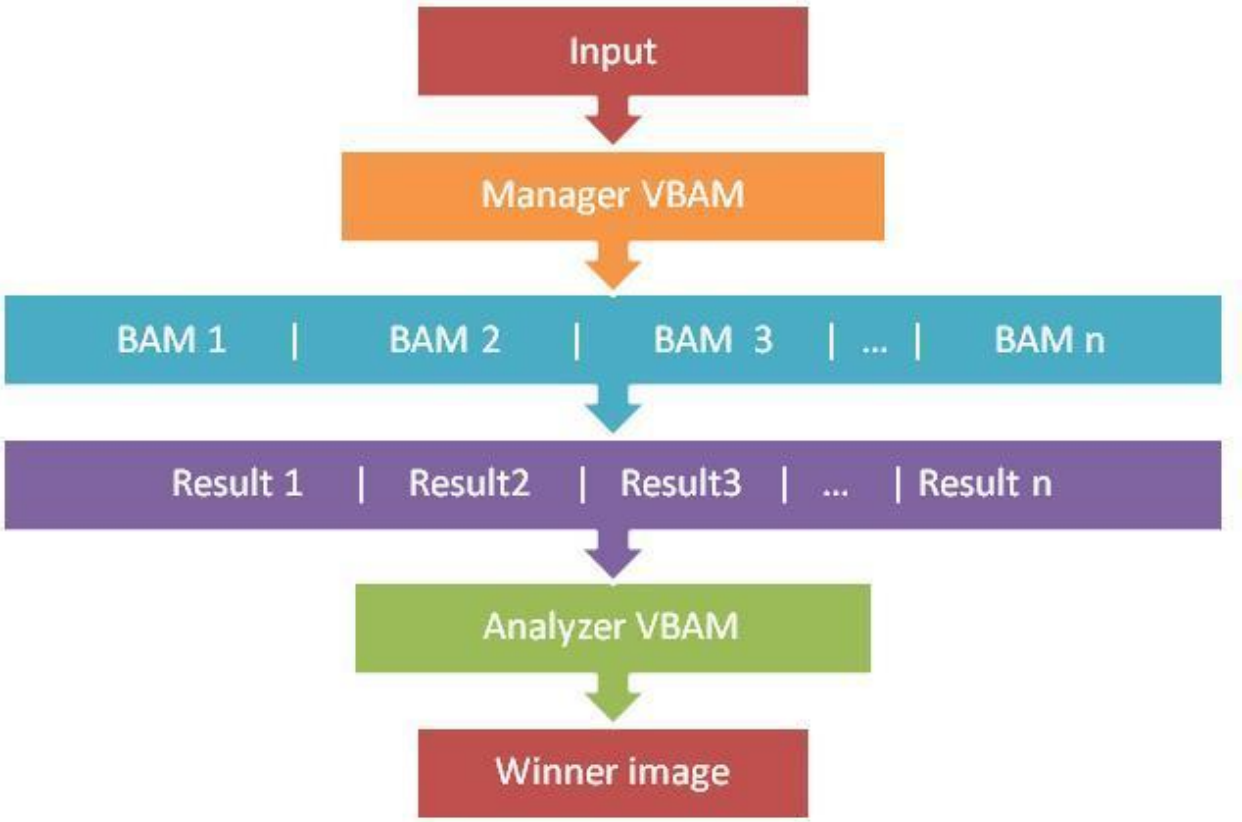
Aplicatia este formata din doua module separate BAM si VBAM.

Va exista un modul VBAM si n module BAM, n >=2.

BAM-urile preiau ca input o imagine si, in urma aplicarii unui algoritm de binarizare, intorc ca output doua imagini: prima imagine reprezinta solutia algoritmului de binarizare, iar a doua reprezinta matricea factorilor de incredere/pixel. Cele n seturi de cate doua imagini sunt evaluate de VBAM, care aplica un sistem de votare, pentru a stabili care algoritm este optim din punct de vedere al imaginii binarizate rezultat.

Daca un algoritm de binarizare se blocheaza sau nu intoarce niciun rezultat, atunci este descalificat, in sensul ca nu va interveni in nici un fel in procesul de votare.

## 5.2 Diagrama de arhitectura




Fiecare rezultat este compus din imaginea binarizata si matricea factorilor de incredere.

## 5.3 Descrierea modulelor

### 5.3.1 BAM

BAM (Binarization Algorithm Module) aplica un algoritm de binarizare asupra imaginii primite ca input si intoarce pentru fiecare pixel o valoare de 0 sau 1 (in imaginea binarizata) si un factor de incredere, cu valori intre 0 si 255 (in matricea factorilor de incredere). Scrie rezultatele conform specificatiilor in fisiere format .PGM, avand numele standardizat astfel: *numeImagine_binarizare.pgm* si *numeImagine_factori.pgm.*

### 5.3.2 VBAM

· Manager – porneste modulele BAM si monitorizeaza timpul de executie si resursele consumate

· Analyzer – preia rezultatele modulelor BAM si aplica algoritmul de votare pentru a determina imaginea castigatoare

## 5.4 Restrictii de implementare

Formatul fisierelor de intrare si de iesire va fi de tip .pgm

Modulele BAM trebuie sa fie executabile, de tip .exe

## 5.5 Interactiunea dintre componente

Utilizatorul porneste modulul VBAM, cu calea catre directorul in care se afla imaginea/imaginile primite ca input, ce urmeaza a fi supuse algoritmilor de binarizare. Modulul VBAM porneste executabilele BAM, care aplica algoritmii de binarizare asupra imaginilor primite ca input si intorc rezultatul, format din imaginea binarizata si matricea factorilor de incredere. Daca timpul de rulare, depaseste un timp predefinit pentru oricare din modulele BAM, atunci modulul BAM este descalificat. Dupa ce se obtine rezultatul modulelor BAM, se porneste algoritmul de votare al modulului VBAM, care are ca rezultat binarizarea cea mai buna si optima a imaginii initiale.

# 6. Modelul interfetei cu utilizatorul

## 6.1 BAM

### 6.1.1 Interfata aplicatiei

Aplicatia nu va oferi o interfata grafica pentru interactiunea utilizatorului cu partea de functionalitate efectiva (implementarea algoritmului). In schimb rularea programului se va realiza exclusiv din linie de comanda.

### 6.1.2 Argumentele interfetei

Algoritmul BAM va primi ca prim argument numele fisierului de intrare (imaginea ce se vrea a fi binarizata), semnalandu-se o eroare in cazul in care fisierul nu exista, nu sunt drepturi de citire, imaginea nu corespunde formatului acceptat. Cel de-al doilea argument este reprezentat de numele fisierului de iesire (imaginea binarizata). Aplicatia va afisa un mesaj de eroare in cazul in care numarul de argumente este mai mic decat 2.

### 6.1.3 Output interfata

Aplicatia va avea drept output imaginea binarizata. De asemenea programul va crea un al doilea fisier de iesire al carui nume va fi de forma <nume_fisier_output>_confidence. In acest fisier vor fi salvate valorile de incredere pentru fiecare pixel din imaginea binarizata (0 - incredere minima, 255 - incredere totala).

## 6.2 VBAM

### 6.2.1 Interfata aplicatiei

La fel ca in cazul BAM aplicatia va fi rulata din linia de comanda.

In urma compilarii va fi creat un executabil  VBAM.exe.

Comanda va avea forma urmatoare: VBAM.exe input_image.png n BAM1.exe BAM2.exe ... BAMn.exe (n = numarul de module BAM).

### 6.2.2 Argumentele interfetei

Algoritmul VBAM va avea cel putin un argument reprezentand numele fisierelor de intrare rezultate din executia algoritmului BAM. In cazul in care nu se primeste niciun argument sau fisierele nu au putut fi deschise se va afisa un mesaj de eroare corespunzator.

### 6.2.3 Output interfata

In urma algoritmului de votare va rezulta un mesaj cu numele imaginii alese drept cea mai buna.

Acest mesaj va fi afisat in consola

# 7. Testarea componentelor

## 7.1 Teste functionalitate corecta

### 7.1.1 VBAM

Pentru asigurarea ca aplicatia functioneaza chiar si in conditii neobisnuite, vor fi create teste in care parametri de intrare vor fi introdusi gresit, primindu-se tipuri de fisiere incompatibile cu specificatiile aplicatiei.

### 7.1.2 BAM

Fiecare componenta va fi testata individual pentru a se asigura corectitudinea algoritmilor folositi.

## 7.2 Testele de performanta BAM

Performanta aplicatiei este puternic influentata de performanta fiecarui element BAM in parte, intrucat, din ratiuni de monitorizare exacta a timpilor de rulare a fiecarui algoritm, acestea se executa intr-un mod secvential.

Bineinteles, aplicatia poate fi implementata ca un mediu multi-threaded, fiecare BAM ruland intr-un thread separat.

## 7.3 Teste de integrare

Dupa executia testelor de corectitudine individuala se va face testarea de integrare, componentele fiind interconectate.