

Manual de utilizare

*Procesare de Imagini - 9 Mai 2019

Catalin Moldovan

Grupa 30235

Universitatea Tehnica din Cluj-Napoca

catalin.moldovan@student.utcluj.ro

Abstract—Acest document descrie modul de utilizare al aplicatiei *Recunoastere pe baza de semnatuiri*

I. ADAUGARE DEPENDINTE

a) Se vor adauga urmatoarele cai in sectiunea **General** din proprietatile aplicatiei:

- \$(SolutionDir)/Libraries/OpenCV-4.1.0/opencv/build/x64/vc15/bin
- \$(SolutionDir)/Libraries/install/include
- \$(SolutionDir)/Libraries/OpenCV-4.1.0/opencv/build/include

b) Se vor adauga urmatoarele cai in sectiunea **Linker** din proprietatile aplicatiei:

- \$(SolutionDir)/Libraries/OpenCV-4.1.0/opencv/build/x64/vc15/lib
- \$(SolutionDir)/Libraries/install/x64/vc15/lib

c) Se vor adauga urmatoarele fisiere in sectiunea **Input** din proprietatile aplicatiei:

- opencv_world410d.lib
- opencv_xfeatures2d410d.lib

II. CREAREA FISIERELOR

In folderul **Semnatura** se vor crea foldere (test1,test2,...) cu imagini de test, utilizate pentru antrenarea algoritmului, dupa urmatoarea regula: **image_XXXX.jpg**, unde XXXX reprezinta numarul imaginii de test (ex. 0001).

In aceeasi locatie se creeaza folderul *input* ce contine imaginea de intrare pentru incadrarea acesteia in una din categoriile create la pasul anterior.

III. PROCESARE IMAGINI DE TEST

In fisierul **Program.cpp**, in variabila **DATASET_PATH** se introduce calea catre fisierul ce contine folderele create la pasul anterior, utilizand doua caractere despartitoare. Variabila **TESTING_PERCENT_PER** contine numarul imaginii introduse in folderul *input*.

Metoda **algoritmSemnatura** va utiliza algoritmul de clasificare K-means. Acesta presupune antrenarea prin imagini de test ce fac parte din anumite categorii, in cazul nostru semnatuiri ale diferitelor persoane. Astfel se vor adauga metode **read-GetImageDescriptors**("test[no]", [semnatuiri], [cat]), unde *no* reprezinta numarul folderului de test, *semnatuiri* numarul de

imagini de test prezente in folderul respectiv si *cat* reprezinta numarul categoriei incepand cu valoare 1 si continuand crescator in functie de numarul de metode adaugate.

Metoda **testData**("input", TESTING_PERCENT_PER, [cat]) reprezinta clasificarea imaginii de intrare cu categoriile de test incluse in variabila *cat*.

Metoda **algoritmHist** va utiliza un algoritm de comparare a imaginilor de test cu cea de intrare prin histograma cu numar de acumulare redus. Se utilizeaza astfel media dintre numarul de pixeli negri de pe orizontala si verticala a fiecarei imaginii de test cu cea de intrare. Se vor adauga numele folderelor de test: (test1,test2,...) in variabila **testImg[no]**, unde *no* reprezinta numarul total de fisiere adaugate.

In pasul urmator se va introduce calea folderelor de test in metoda **processedImage** din variabila **imgTest** si calea imaginii de intrare din variabila **srcInput**.

IV. RULARE PROGRAM

Rularea programului se va efectua in modul **Debug** prin *Local Windows Debugger*.

In procesul algoritmului K-means se vor citi, de la dectă și se vor transforma imaginile de test, urmând ca apoi să se genereze histogramele și antrenarea SVM. Rezultatul final este redat prin obținerea valorii de 0% sau 100% a acuratetii clasificării pentru fiecare categorie de test. Pentru fiecare proces este calculat timpul de execuție în secunde.

In procesul algoritmului de identificării semnăturii prin histogramă se redă în consolă procentajul pentru fiecare categorie. În final procentajul cel mai ridicat, cu condiția ca acesta să depășească 60% reprezintă potrivirea semnăturii primite cu categoria din care face parte.

REFERENCES

- [1] "DescriptorMatcher::match", website: https://docs.opencv.org/2.4/modules/features2d/doc/common_interfaces_of_descriptor_matchers.html
- [2] "SVM::predict", website: https://docs.opencv.org/3.1.0/d1/d73/tutorial_in_troduction_to_svm.html
- [3] "cv::xfeatures2d::SIFT", website: https://docs.opencv.org/3.4/d5/d3c/class_cv_1_1xfeatures2d_1_1SIFT.html
- [4] "kmeans", website: https://docs.opencv.org/3.4/d1/d5c/tutorial_py_kmeans_opencv.html