

Recunoastere pe baza de semnături

*Procesare de Imagini - 9 Mai 2019

Catalin Moldovan

Grupa 30235

Universitatea Tehnica din Cluj-Napoca

catalin.moldovan@student.utcluj.ro

Abstract—Acest document descrie doi algoritmi eficienți ce pot fi utilizați pentru verificarea semnăturii unei persoane. Metoda partitionării imaginilor de test prin histograma cu un număr de acumulatori redus și algoritmul de clasificare K-means pe baza de SVM. Algoritmul poate fi utilizat în diferite companii, banci, pentru validarea documentelor și a contractelor.

Index Terms—Preprocesare, Thresholding, Histograma, KNN, K-means, Detectarea varfurilor, Distanța Euclidiană

I. INTRODUCERE

Semnatura reprezintă o metoda de identificare a unei persoane, ce conține diferite caractere ce nu pot fi întotdeauna citibile. Prin evoluția tehnologiei, a evoluat în același timp și modul de verificare a semnăturii unei persoane, crescând probabilitatea de recunoaștere. Analiza unei semnăturii pe baza unei imagini se efectuează ca imagine întreagă și nu ca litere și cifre puse împreună. Biometria face referire la autentificarea unei persoane în funcție de factori psihologici și comportamentali. Aceasta face diferență dintre indivizi. În această lucrare se va pune problema pe biometria scrisului și reprezentarea unei metode de verificare a semnăturii folosind procesare de imagini. Pentru verificarea unei semnături, se vor utiliza un set de semnături obținute de către posesor în trecut. Prin efectuarea unor comparații a unor factori, se validează integritatea semnăturii.

A. Metode de verificare

Verificarea unei semnături poate fi împărțită în două categorii: online și offline.

a) *Online*: Verificarea semnăturii online reprezintă o metoda de recunoaștere a scrisului în decursul scrierii. Există diferiți factori, precum timpul petrecut cu varful pixului pe suprafață, forța de apăsare, etc. Se pot extrage diferite elemente ce pot caracteriza posesorul semnăturii. Aceasta metoda duce la o acuratețe mai bună, deoarece percepțiile emoționale și de presiune sunt foarte greu de imitat.

b) *Offline*: În cazul semnăturilor offline este necesară scanarea imaginii sau fotografierea ei, iar apoi folosirea unei aplicații digitale pentru detectarea autenticității.

II. METODE DE VERIFICARE

O metoda de identificare a semnăturii este antrenarea sistemului cu un număr de 6 semnături de test, pe baza algoritmului “Backpropagation” [1].

O altă metoda de identificare este printr-un proces de detectare a varfurilor prin algoritmul Canny (Secțiunea V). În final se va folosi distanța Eclidiană (Secțiunea VI) pentru clasificarea imaginii cu cele de test [2].

III. IDENTIFICAREA SEMNATURII

A. Preprocesarea imaginii

Pașii necesari pentru transformarea unei imagini sub o formă convenabilă sunt următorii:

- imagine grayscale
- binarizare imagine
- decupare imagine
- normalizare imagine

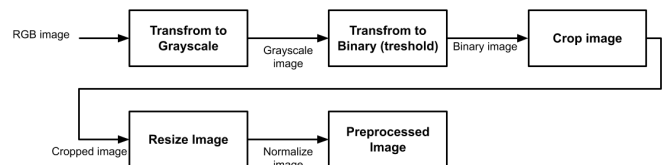


Fig. 1. Pași preprocesare

Primul pas reprezintă conversia imaginii RGB într-o imagine grayscale. Prin acest proces se va reduce complexitatea și timpul de execuție al sistemului. Este mult mai ușor de executat operații pe o imagine alb-negru, decât pe o imagine color. Este necesar un proces de binarizare a imaginii pentru îndepărtarea zgomotului. Următorul pas este reprezentat de un algoritm de decupare. Decuparea constă în separarea regiunii de interes față de imaginea completă. Prin aceasta se vor îndepărta pixelii albi, ceea ce va reduce mai mult timpul de procesare.

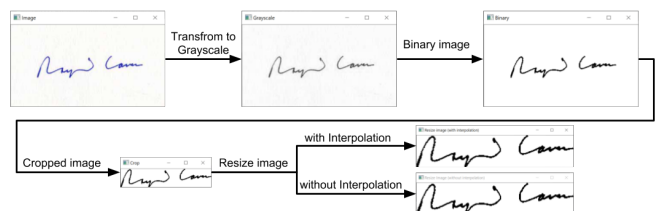


Fig. 2. Preprocesare imagine

Normalizarea semnăturii va elimina diferențele de dimensiune a imaginii curente pentru a corespunde exact cu semnăturile de test din baza de date. Astfel pentru diverse comparații, rezultatele vor fi mai ușor vizibile, Fig. 2.

B. Propunerea sistemului

Diagrama bloc propusă sistemului este prezentată în Fig. 3.

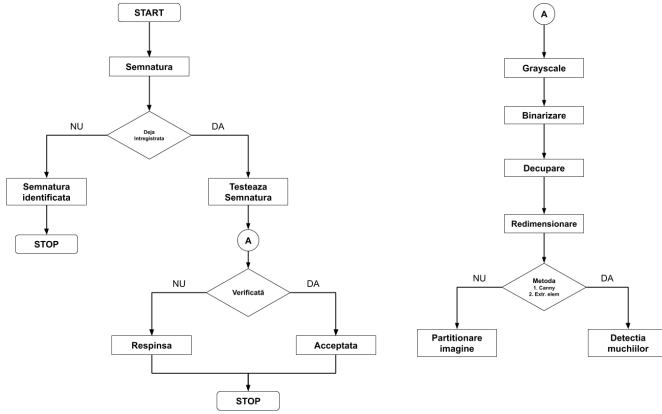


Fig. 3. Diagrama Bloc

IV. METODA PARTITIONARII

A. Extragerea elementelor

O metodă de verificare a semnăturii este compararea anumitor elemente ce țin de numărul de pixeli pe orizontală și verticală a imaginii decupate. Imaginea decupată va duce la eliminarea diferitelor proporții ale semnăturii. În funcție de calitatea imaginii, aceasta este împartită într-un număr redus de acumuloare, Fig. 4.

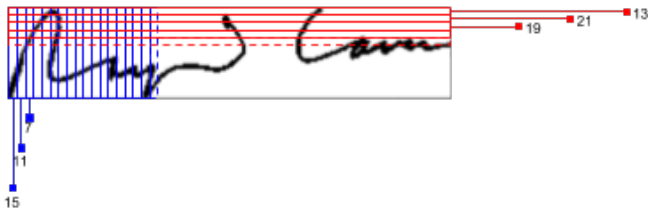


Fig. 4. Partitionare semnatura

După finalizarea parcurgerii imaginii, putem concepe o histogramă bazată pe numărul de pixeli pe orizontală și pe verticală a imaginii decupate, împartind lungimea și înălțimea imaginii în m , respectiv n părți egale, iar apoi să numărăm pixelii care aparțin fiecărei părți, Fig. 5.

$$H(\text{height}) = N_h, h \in [0 \dots m] \quad (1)$$

$$H(\text{width}) = N_w, w \in [0 \dots n] \quad (2)$$

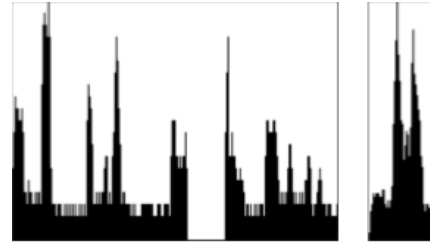


Fig. 5. Histograma pixeli pe orizontală și verticală

B. Clasificare

Această secțiune prezintă modul de identificare a semnăturii persoanei. Se vor efectua comparații succesive a numărului de pixeli pe orizontală și verticală a imaginii decupate cu cele de test. Pentru fiecare comparație cu o imagine de test va rezulta o probabilitate, ce semnifică rata de identificare a posesorului semnăturii. Media probabilităților fiecărui set de semnături de test, respectiv probabilitatea cea mai ridicată și cu un procent mai mare de 60%, va reprezenta o potrivire a semnăturilor. Un exemplu este prezentat în Tabelul I.

TABLE I
SET DE PROBABILITĂȚI

Probabilitate Imagine	Comparare semnături baza de date		
	Prob. Test 1	Prob. Test 2	Prob. Test 3
65%	45%	76%	57%
25%	24%	36%	17%
82%*	75%	91%	88%
34%	33%	45%	27%

*Setul de imagini cu probabilitatea cea mai ridicată.

V. METODA CANNY

Metoda Canny se bazează pe calculul gradientului imaginii și în plus permite maximizarea raportului semnal zgomot pentru o detecție corectă, o localizare bună a punctelor de muchie și minimizarea numărului de răspunsuri pozitive la o singură muchie singulară, Fig. 6.

Pășii metodei Canny [3]:

- Filtrarea imaginii cu un filtru Gaussian pentru eliminarea zgomotului.
- Calculul modulului și direcției gradientului.
- Suprimarea non-maximelor modulului gradientului.
- Binărizarea adaptivă a punctelor de muchie și prelungirea muchiilor prin histereză.

A. Filtrarea imaginii cu un filtru Gaussian

Zgomotul din imagine este o informație de “frecvență înaltă” care se suprapune peste imaginea originală. Aceasta induce apariția unor puncte de muchie false. Zgomotul inerent procesului de achiziție al imaginilor are un model Gaussian și se poate elimina cu un filtru omonim.

B. Calculul modului si directiei gradientului

Calcularea modului gradientului (Ecuatia 3) si directiei gradientului (Ecuatia 4) presupune alocarea a câte unui buffer temporar de dimensiunea imaginii și inițializarea elementelor lor.

$$|\nabla f(x, y)| = \sqrt{(\nabla f_x(x, y))^2 + (\nabla f_y(x, y))^2} \quad (3)$$

$$\theta(x, y) = \arctg\left(\frac{\nabla f_y(x, y)}{\nabla f_x(x, y)}\right) \quad (4)$$

Componentele orizontale $\nabla f_x(x, y)$ si respectiv verticale $\nabla f_y(x, y)$ se calculeaza prin convolutia imaginii prin nucleul Sobel (Ecuatia 5).

$$\nabla f_x = f(x, y) * \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad (5)$$

C. Suprimarea non-maximelor modului gradientului

Suprimarea are drept scop subtierea muchiilor prin pastrarea doar a punctelor de muchie care au modulul maxim al gradientului de-a lungul directiei de variatie a intensitatii, de-a lungul directiei gradientului. Astfel se cuantifica directiile gradientului in 4 intervale de câte 45° . In final se obtine o imagine in care intensitatea pixelilor este egala cu valoarea modului gradientului in acel punct.

D. Binarizarea adaptiva a punctelor de muchie si prelungirea muchiilor prin histereza

a) *Binarizarea adaptiva*: Incearca sa extraga un numar relativ constant de puncte de muchie pentru o dimensiune data a imaginii. In acest fel se compenseaza iluminarea si contrastul diferit intre imagini. (Ecuatia 6), unde parametrul p ia valori intre 0.01 și 0.1. Totusi nu garanteaza completitudinea muchiilor, deoarece anumite parti umbrite ale obiectului sau prezenta zgomotelor pot afecta procesul de detectie. Rezultatul va fi o imagine cu foarte multe muchii fragmentate.

$$NrPctMuchie = p * (NrPixeli - NrPixeliGradNul) \quad (6)$$

Algoritm:

- 1) Se calculeaza histograma de intensitati a imaginii.
- 2) Se calculeaza numarul de pixeli diferiti de zero care nu vor fi puncte de muchie.
- 3) Gasim valoarea intensitatii sub care se gasesc numarul de non muchii pixeli.

b) *Extinderea muchiilor prin histereza*: Se impune o tehnica de prelungire a muchiilor. Muchiile obtinute prin binarizarea initiala se cauta a se prelungi cu muchii mai putin clare, care nu trec testul binarizarii cu acest prag, dar ar putea sa apara ca muchii la un prag mai scazut.

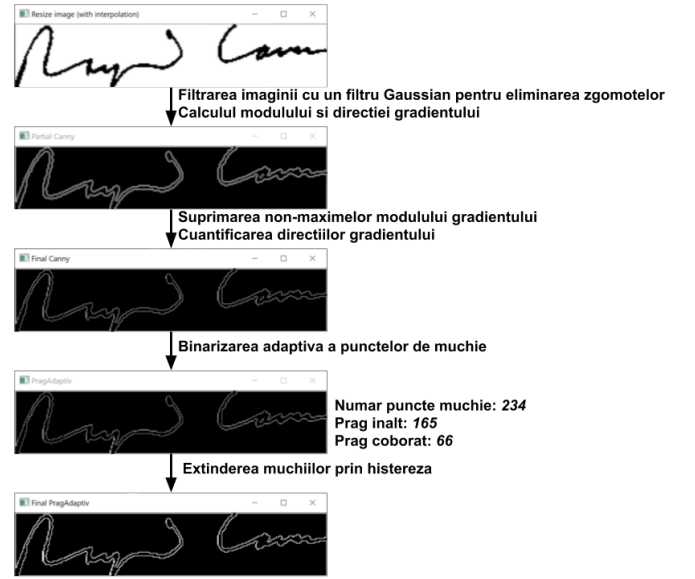


Fig. 6. Detectia punctelor de muchie

VI. K-NEAREST NEIGHBOURS

KNN este o metoda de clasificare bazata pe instante. Algoritmul presupune ca imaginea este plasata intr-un spatiu n-dimensional. Fiecare instanta de test este clasificata in functie de cele mai apropiate k exemple din imaginile de test, fara a incerca sa se determine functia de distributie pe baza careia instantele sunt plasate in spatiul n-dimensional [4].

Pentru a gasi si extrage din exemplele de test, este nevoie de o metoda de a compara cat de similare sunt doua instante. Pentru aceasta se foloseste distanta euclidiană. Urmatorul pas este sa se calculeze distanta dintre imagine si toate exemplele de test. In ultimul pas se determina care sunt cele mai apropiate k instante din setul de imagini de test si pe baza acestor instante similare se decide care este cea mai probabila clasificare a acestora. Acest lucru se realizeaza prin votarea intre cele mai similare k instante, Fig. 7 [5].

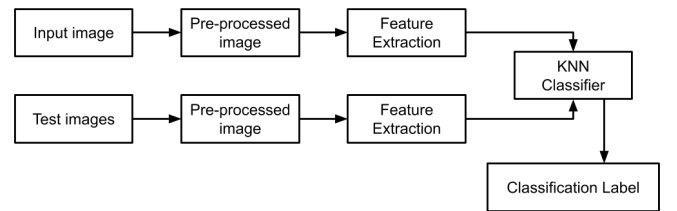


Fig. 7. Diagrama bloc a metodei propuse utilizand KNN

A. Distanța Euclidiană

Distanța euclidiană reprezintă distanța geometrică între două puncte în spațiul bidimensional definită ca linia dreaptă care le unește (Ecuatia 7). În alte cuvinte, reprezintă distanța dintre două puncte ce poate fi măsurată cu un liniar. Aceasta este data de teorema lui Pitagora.

Folosind aceasta formula ca distanta, spatiul euclidian devine spatiu metric. Astfel putem genera valori exacte prin potrivirea imaginii date cu semnăturile de test [6].

$$d_E(P_1, P_2) = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} = ||P_1 - P_2|| \quad (7)$$

VII. K-MEANS

Pentru compararea imaginilor de test cu cea de intrare se va folosi un algoritm de clasificare a celor mai apropiati vecini, apoi vom trece la clasificarea liniara utilizand SVM (support vector machines). Algoritmul 'bag of words' va realiza o clasificare bazata pe histograma de frecventa a anumitor caracteristici. Astfel se vor compara detectori, Laplacian, descriptori si functii kerneli SVM, Fig. 8. [7]

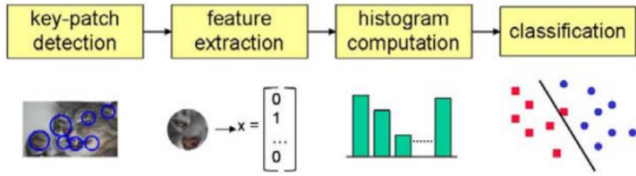


Fig. 8. Procesul de clasificare K-means

Pentru estimarea distantei functiei kernel, se va forma o *image de semnatura* (Ecuatia 8).

$$S = ((t_1, \mathbf{m}_1), \dots, (t_m, \mathbf{m}_m)) \quad (8)$$

Urmatorul pas reprezinta investigarea a doi kerneli diferiti pentru compararea *imaginii de semnatura* prin EMD (earth mover's distance) (Ecuatia 9), unde f_{ij} reprezinta fluxul si $d(\mathbf{m}_i, \mathbf{m}'_j)$ distanta Euclidiană (Ecuatia 7).

$$EMD(S, S') = \frac{\sum_i \sum_j f_{ij} d(\mathbf{m}_i, \mathbf{m}'_j)}{\sum_i \sum_j f_{ij}} \quad (9)$$

A doua distanta calculata va masura modul in care doua semnături au fost generate prin procesul de consistenta intamplatoare, apoi aceasta fiind convertita in kernel SVM folosind un kernel Gaussian (Ecuatia 10), unde A reprezinta parametrul de scalare stabilit la deistanta medie dintre imaginile de antrenament.

$$K(S, S') = \exp\left(-\frac{1}{A} D(S, S')\right) \quad (10)$$

Imaginile de test vor fi impartite in categorii unice, fiecare categorie reprezentand un individ, avand un anumit numar de semnături de test pentru antrenarea programului. In functie de descriptorii extrasi din fiecare imagine se vor compara imaginile prin algoritmul de grupare K-means. [8]

Scopul metodelor de grupare (clusterizare) este de a partitiona o multime de obiecte in grupuri diferite, unde instantele dintr-un grup sunt similare intr-un anumit sens, Fig. 9.

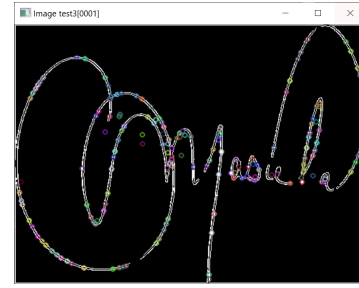


Fig. 9. Descriptorii extrasi din semnatura

Algoritmul k-means primeste ca valori de intrare o lista de puncte $X = \{x_i, i = 1 : n\}$. Fiecare punct este d-dimensional $x_i = (x_{i1}, x_{i2}, \dots, x_{id})$. Obiectivul metodei k-means este gruparea punctelor în K multimii notate cu $S = \{S_k | k = 1 : K\}$. Centroidul care reprezinta submultimea k este notata cu m_k . Gruparea datelor trebuie realizata astfel incat sa fie minimizata functia obiectiv (Ecuatia 11), unde $dist(., .)$ reprezinta distanta Euclidiană in spatiul d-dimensional (Ecuatia 7).

$$J(X, S) = \sum_{k=1}^K \sum_{x \in S_k} dist(x, m_k) \quad (11)$$

In procesul algoritmului K-means se vor citi, decata si transforma imaginile de test, urmand ca apoi sa se genereze histogramele si antrenarea SVM. Rezultatul final este redat prin obtinerea valorii de 0% sau 100% a acuratetii clasificarii pentru fiecare categorie de test. Pentru fiecare proces este calculat timpul de executie in secunde. Un exemplu este prezentat in Tabelul II.

TABLE II
SET DE PROBABILITATI

Categorii	Rezultat
Categorie 1 de imagini de test	0%
Categorie 2 de imagini de test	100%
Categorie 3 de imagini de test	0%
Categorie 4 de imagini de test	0%

VIII. COMPARATIE ALGORITMI UTILIZATI

Metoda de verificare a semnăturii prin compararea histogramei a numarului de pixeli de pe verticala si orizontala, folosind un numar redus de acumuloare, este utilizata pentru rezultate probabilistice pentru semnatura de intrare. Se obtine un procentaj pentru fiecare categorie parcursa in relatie cu imaginea de intrare. In schimb, algoritmul Kmeans este utilizat pentru rezultate mult mai precise pe baza antrenarii algoritmului cu imagini de test. Pe baza anumitor descriptori se creeaza un pattern, in urma careia se incadreaza imaginea intr-o anumita categorie.

a) *Avantaje/Dezavantaje algoritm partitionare:*

- Cu cat numarul de acumuloare este mai crescut, cu atat algoritmul de partitionare este mai precis in obtinerea rezultatelor.

- Prin introducerea unei imagini ce nu face parte din nici o categorie, algoritmul nu va încadra imaginea într-o anumită categorie.
- Algoritmul este rapid în execuție, iar datele sunt prezente imediat după parcurgerea unei categorii.

b) Avantaje/Dezavantaje algoritmului Kmeans:

- Algoritmul necesită un număr mare de imagini de test pentru o acuratețe mai bună.
- Prin introducerea unei imagini ce nu face parte din nici o categorie, algoritmul este nevoit să încadreze semnătura într-o categorie cât mai apropiată de pattern-ul creat.
- Algoritmul necesită un timp de execuție destul de mare pentru antrenare.

IX. CONCLUZII

Acest document oferă o modalitate de verificare eficientă și simplă a unei semnături în mediul offline. Algoritmul a fost testat pe un set de semnături de test, iar rezultatele obținute au fost în cea mai mare parte valide.

REFERENCES

- [1] Sikander Hans, Sachin Gupta, "Algorithm For Signature Verification System" Department of Electrical Instrumentation Engineering, Thapar University, Patiala, India, April 2012
- [2] Mohan Mandaogade, Saurabh, Vishal Mhaske, "Handwritten Signature Verification And Recognition Using ANN", MPGI National Multi Conference 2012, 1892, 7-8 April 2012
- [3] Radu Gabriel Danescu, "Image Processing", Laboratory 11: Edge detection
- [4] Costin-Gabriel Chiru, Stefan Trausan-Matu, Traian Rebedea, "O îmbunătățire a performanțelor algoritmului KNN în sistemele de recomandare pe web" Interacțiune Om-Calculator, Universitatea Politehnica București, 2008
- [5] P.A.Charde, S.D.Lokhande, "Classification Using K Nearest Neighbor for Brain Image Retrieval", International Journal of Scientific & Engineering Research, Volume 4, Issue 8, August-2013, 760-765, ISSN 2229-5518
- [6] Isha, Pooja, Varsha, "Offline Signature Verification based on Euclidean distance using Support Vector Machine", International Journal of Advanced Engineering, Management and Science (IAEMS), Vol-2, Issue-8, Aug-2016, ISSN: 2454-1311
- [7] Richard Szeliski, "Computer Vision: Algorithms and Applications", September 3, 2010, Chapter 5.3.1 "K-means and mixtures of Gaussians" pages 289-292, Chapter 14.4.1 "Bag of words" pages 697-701
- [8] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A K-Means Clustering Algorithm", Vol. 28, No. 1 (1979), pages 100-108
- [9] "DescriptorMatcher::match", website: https://docs.opencv.org/2.4/modules/features2d/doc/common_interfaces_of_descriptor_matchers.html