Recunoastere faciala – documentatie

Pentru a face expresorul de cafea mai inteligent, prin a-l invata cine este in fata lui si daca este sau nu oboist, vom folosi, pentru partea SW, un script scris in limbajul de programare **python**. Acesta va fi responsabil pentru **recunoastere si identificare faciala**.

Deci vom avea doua “**module**” unul specializat pe recunoastere si identificare iar altul pe antrenarea retelei neuronale.

Recunoasterea este direfita de identificarea faciala din mai multe puncta de vedere, pe care le vom discuta in curand.

Pentru realizarea acestor lucruri avem nevoie de **Pyton3**, **OpenCV** (**O**pen-source **C**omputer **V**ision) cu toate dependentele aferente acestuia si modulul **dlib**.

**OpenCV**

OpenCV este un modul(librarie) open-source pentru algoritmi de “vedere computerizata”. Cu ajutorul acestul modul python esterealizata partea de recunoastere si identificare faciala astfel:

* O multime de imagini(positive si negative) representative
* Pe baza multimii de imagini se vor antrena o retea neuronala care va fi capabila sa detecteze sis a identifice fata unei persoane

Imaginile positive trebuie sa contina, de preferat, doar obiectul de interes, sau in cazul nostru, o fata. Iar cele negative opusul.

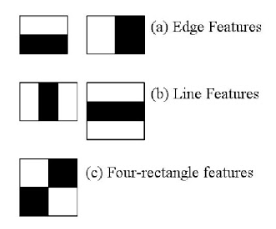
Scriptul care face diferentierea intre un raspuns pozitiv si un raspuns negativ se numeste clasificator (**classifier**). Un clasificator este antrenat, de regula, pe mii sau zeci de mii de imagini positive/negative. In cazul nostru nu va fi nevoie de atat de multe exemple deoarece OpenCV ne pune la dispozitie un clasificator pre-antrenat pentru recunoastere, noi fiind nevoiti doar sa ne antrenam un identificator.

De interes pentru noi, din OpenCV sunt clasificatorii in cascada de tip **HAAR** si **LBP**. Clasificatorii nu au nevoie de informatia continuta in imagine sub forma de culoare, de aceea toate imaginile care vor fi prelucrate de acestea vor fi alb-negru (**grayscale**).

In cele ce urmeaza vom incerca, pe scurt sa explicam diferentele, avantajele si rezavantajele celor doua tipuri de clasificatori.

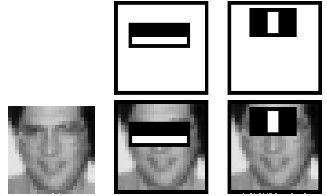
**Clasificatorul in cascada de tip HAAR**

Clasificatorul de tip Haar are la baza un algoritm(creeat de Paul Viola si Michael Jones in 2001)in care, practic, cauta si compara zone din imagini. Din imagine se vor extrage caracteristici cu ajutori caracteristicilor de tip haar reprezentate mai jos.



Img. 1 – Caracteristicile de tip haar

In imaginile primite de la camera, se definesc zeci sau chiar sute de mii de zone(proportional cu rezolutia) cu ajutorul caracteristicilor de tip haar, dar clasificatorul in cascada te tip haar disponibil verifica fiecare caracteristica pe zone si daca informatia obtinuta nu este relevanta, algoritmul ignora complet zona repectiva si trece mai departe la o noua zona din imagine. Acest process de “filtrare” a zonelor se numeste **Adaboost**



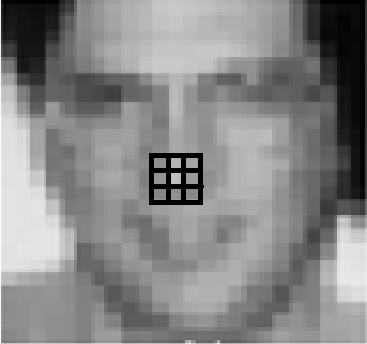
Img. 2 – Caracteristicile de tip haar suprapuse pe imaginea de interes

In imaginea de mai observam cum toate caracteristicile te tip haar sunt aplicate imaginii de interes. Pentru fiecare caracteristica se gaseste un prag decisive, dupa care se decide in final daca imaginea este pozitiva sau negativa.

**Clasificatorul de tip LBP**

Clasificatorul LBP(**L**ocal **B**inary **P**attern) sau modelul local binar este un descriptor visual/textural care proceseaza imaginea in modul urmator:

* Imparte imaginea(matricea bidimensionala) in subcategorii(submatrici) de 3x3 pixeli



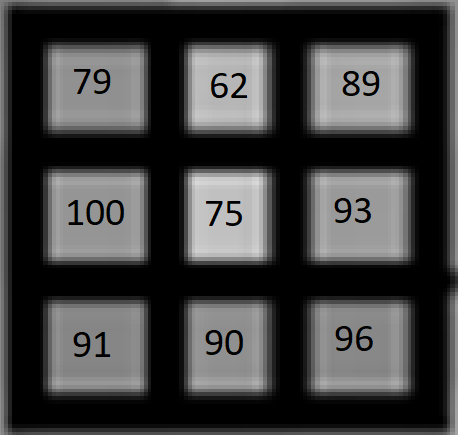
Img 3. Imagine impartita in sectiuni de 3x3

* Selecteaza zona de 3x3



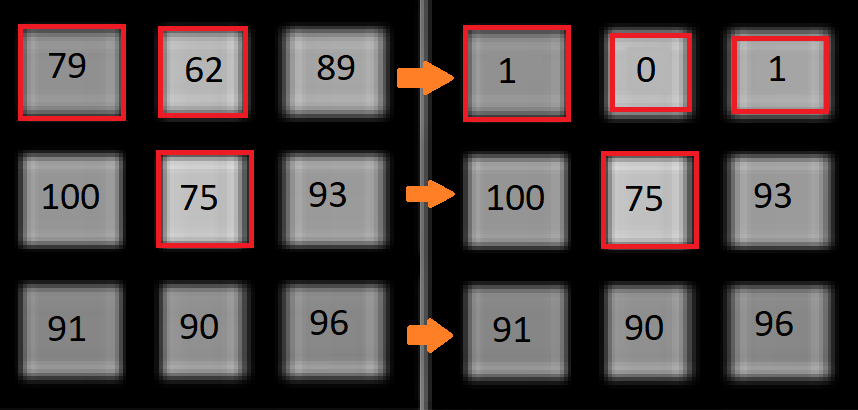
Img 4. Submatricea de 3x3 selectata de algoritm

* Compara valorile(0-255) celor 8 pixeli exterior cu valoarea pixelului din mijlocul matricei in sensul acelor de ceasornic



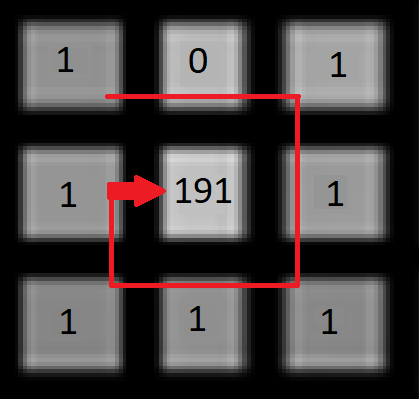
Img 5. Compararea valorilor pixelilor

* Daca valoarea unui pixel raportata lavaloarea pixelului central este mai mica, atunci valoarea acestuia se scuprascrie cu o oua valoare, 0 si 1 daca aceasta este mai mare.



Img 6. Suprascrierea valorilor bitilor cu echivalentul in binar rezultat din comparative

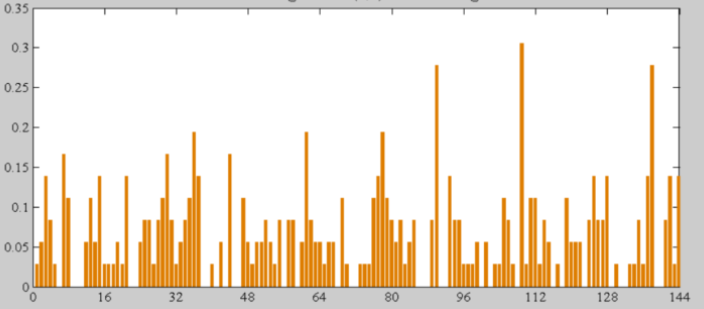
* Valorile obtinute in final vor fi considerate un nou sir binary iar valoarea acestuia in decimal va lua locul valorii pixelului din mijlocul matricii



Img. 7. Valorile obtinute in urma comparatiei

(1011 1111)2=(191)10

* Se repeat pasii de mai sus pentru fiecare portiune de 3x3 din imagine pana cand toata suprafata imaginii va fi acoperita.
* La final toate valorile submatricilor de 3x3 se vor converti intr-o histogramam si se concateneaza pentru a forma un vector ce descrie o trasatura(feature)



Img 8. Histograma valorlor blocurilor de 3x3 pixeli din imagine

**Haar vs.LBP**

|  |  |  |
| --- | --- | --- |
|  | Avantaje | Dezavantaje |
| Haar | 1. Acuratete de detective mare  2.Rata scazuta de detectare a pozitivelor false | 1.Complexitate computationala crescuta  2.Timp necesar pentru antrenare crescut  3.limitari mari in ceea ce priveste iluminarea |
| LBP | 1. Complexitate scazuta  2.Rapiditate foarte mare  3.Robustete in detectarea fetelor indifferent de puterea iluminarii | 1 Acuratete scazuta  2. Rata de detectare a pozitivelor false crescuta |

**Concluzii**

In final am decis sa folosim ceea ce este mai bun din fiecare, deci o combinatie de haar (Clasificatorul de tip cascada Haar din modulul OpenCV) si LBP(antrenat de noi pe fetele noastre) pentru a obtine performante si viteze acceptabile.