Universitatea Tehnica din Cluj-Napoca
Facultatea de Automatica si Calculatoare
Departamentul Calculatoare

Recunoașterea plăcuței de înmatriculare

Student: Iulian-Laurențiu Borșa, 30238

Indrumator: Asist. Radu Bob

1. Introducere

- Descrierea problemei abordate

Identificarea sau recunoasterea automata a placutelor de inmatriculare, cunoscuta ca ANPR (Automatic number-plate recognition) este o tehnologie care utilizeaza recunoasterea caracterelor din placutele de inmatriculare surprinse in imagini, pentru a obtine date privind localizarea vehiculelor. Aceasta tehnologie este deseori utilizata la parcarile automatizate, la intrarea in spatii private, pe drumuri publice, in zone de taxare auto.

- Contextul problemei

Problema consta in surprinderea imaginilor cu placute de inmatriculare a masinilor si detectarea, cu incadrarea intr-un dreptunghi a suprafetei placutei. Problema de rezolvat trebuie sa aiba ca rezultat o imagine color cu placuta de inmatriculare incadrata, o imagine binara numai cu caracterele obtinute dupa aplicarea algoritmilor de detectie si a textului sub forma de caractere.

- Motivare

Principalul motiv pentru care am ales acest proiect este libertatea de a alege un algoritm specific pentru problema de rezolvat. De asemenea, este un bun start pentru un proiect personal de pe care il pot utiliza la un dispozitiv de deschidere a unei porti. In plus, consider ca este un bun proiect care sa ma ajute sa trec prin toate cunostintele dobandite la laboratorul de procesare de imagini.

2. Studiul bibliografic

- Descrierea sumara a unui algoritm din literatura de specialitate

Există șapte algoritmi principali pe care software-ul ii necesită pentru a identifica o plăcuță de înmatriculare:

- Localizarea placutei - responsabila pentru găsirea și izolarea plăcii de pe imagine.
- Orientarea plăcii și redimensionarea - compensează oblicul plăcii și ajustează dimensiunile la dimensiunea necesară.
- Normalizare - reglează luminozitatea și contrastul imaginii.
- Segmentarea caracterelor - găsește caracterele individuale de pe plăcuțe.
- Recunoaștere optică a caracterelor.
- Analiza sintactică / geometrică - verificați caracterele și pozițiile în funcție de regulile specifice fiecărei țări.
- Medierea valorii recunoscute pe mai multe câmpuri / imagini pentru a produce un rezultat mai fiabil sau mai sigur. Mai ales că orice imagine singulară poate conține o lumină reflectată, poate fi parțial acoperită sau alt efect temporar.

Complexitatea fiecăreia dintre aceste subsecțiuni ale programului determină exactitatea sistemului. În timpul celei de-a treia faze (normalizare), unele sisteme utilizează tehnici de

detectare a marginilor pentru a mări diferența de imagine între litere și suportul plăcii. Un filtru median poate fi, de asemenea, utilizat pentru a reduce zgomotul vizual al imaginii.

3. Metoda propusa

- Descrierea metodei propuse

Metoda software propusa pentru detectarea placutei de inmatriculare este structurata pe functii. Functiile aplicatiei sunt urmatoarele:

- void set_name(char* source, char* destination, const char* extension);

Aceasta functie este folosita pentru modificarea numelui fisierului destinatie. Atunci cand citim imaginea, obtinem un path catre locul in care este salvata imaginea. Mai departe, la acest path ii este adaugat, inainte de extensie, un label care reprezinta tipul de imagine salvata (ex. _binary.bmp, _grayscale.bmp)

- void color_to_grayscale(Mat &source, Mat &destination);

Aceasta functie este folosita pentru transformarea imaginii din color in grayscale.

- void grayscale_to_binary(Mat &source, Mat &destination, int threshold);

Aceasta functie este folosita pentru transformarea imaginii din grayscale in binary.

- void get_labels(Mat image, Mat &labels);

Functia “get_labels” implementeaza un algoritm de etichetare. Acest algoritm reprezinta o metodă directă pentru etichetare, care se bazează pe traversarea în lățime a grafului format de imaginea binară. Primul pas este inițializarea matricei de etichete cu valoarea zero pentru toți pixelii, indicând faptul că inițial totul este neetichetat. Apoi, algoritmul va căuta un pixel de tip obiect care este neetichetat. Dacă acest punct este găsit, el va primi o etichetă nouă, pe care o va propaga vecinilor lui. Vom repeta acest proces până când toți pixelii obiect vor primi o etichetă.

- void open_image(char* source_name, char* destination_name);

Acesta este in principiu corpul intregului program. Mai exact, aici se deschide prima imagine color, care devine mai departe subiect pentru aplicatie. Aceasta imagine este redimensionata pentru optimizarea vitezei programului, la lungimea/latimea de maxim 720p. Mai departe, imaginea color este transformata in imagine grayscale, iar apoi, imaginea grayscale este transformata in imagine binara. Imaginea binara este trimisa mai departe spre etichetare, dupa care se prelucreaza matricea de etichete pentru a reduce numarul de obiecte care nu corespund asteptarilor. In cele din urma, se aplica algoritmul lui Canny de detectie a conturului pentru imaginea binara si astfel se obtine un vector de contururi, dintre care sunt selectate tot cate patru varfuri care formeaza cate un dreptunghi. Aceste dreptunghiuri sunt filtrate dupa raportul $3.5 < \text{width/height} < 5.5$. Acest interval ar trebui sa pastreze doar etichetele care au forma unei placute de inmatriculare.

- void get_image(char* source_name, char* destination_name);

Rezultatul acestei metode este o matrice care reprezinta intersectia dintre imaginea color sursa si matricea de etichete care contine placuta de inmatriculare. Astfel, get_image functioneaza ca o masca pentru imaginea subiect si inlocuieste tot ce nu face parte din placuta de inmatriculare cu alb.

- void get_text(char* source_name);

Cu aceasta metoda se face o ultima prelucrare a imaginii mici, adica a placutei de inmatriculare. Se incearca eliminarea a cat mai multe obiecte care nu sunt caractere.

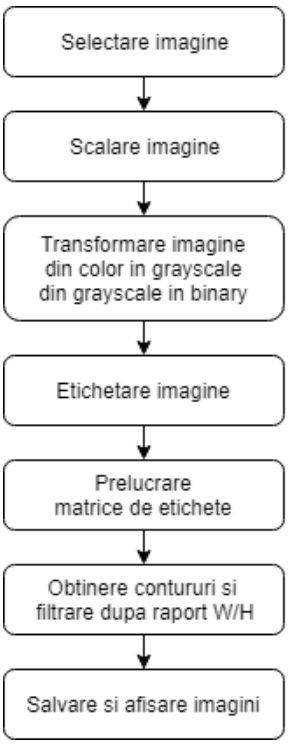
- void show_result(char* source_name);

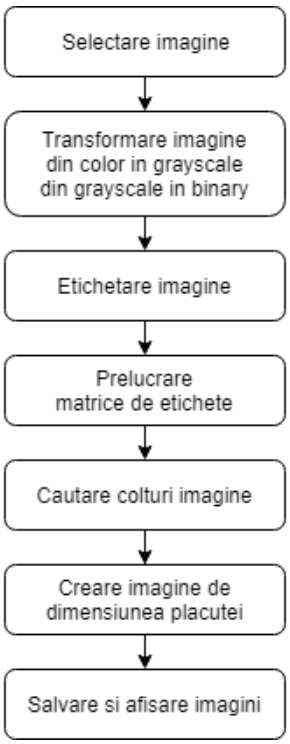
Aceasta este ultima metoda apelata, care are rolul de a aplica un contur/chenar peste placuta de inmatriculare, folosind imaginea sursa si coordonatele punctelor care descriu dimensiunea imaginii din get_image.

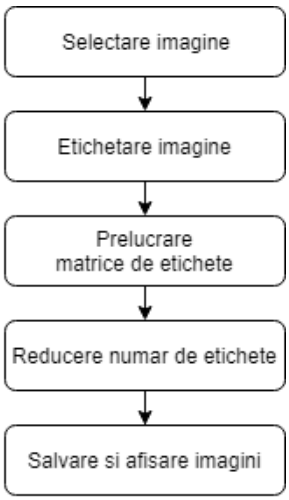
- float euclidian_distance(Point2f p, Point2f q);

Distanta euclidiana reprezinta distanta dintre doua puncte date. Aceasta distanta este utilizata pentru a gasi raportul dorit, intre lungimea si latimea obiectului final etichetat, care reprezinta placuta de inmatriculare.

In continuare este prezentat blocul celor mai importante 3 functii, in ordinea: open_image(params), get_image(), get_text().



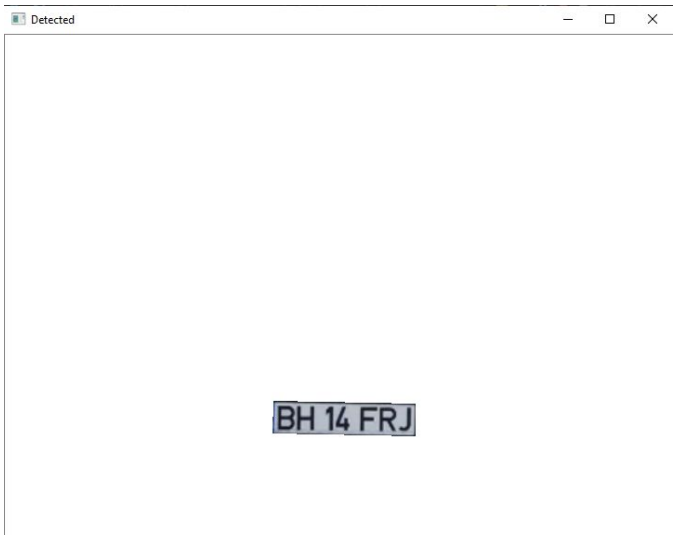




4. Rezultate experimentale

Imagine sursa:

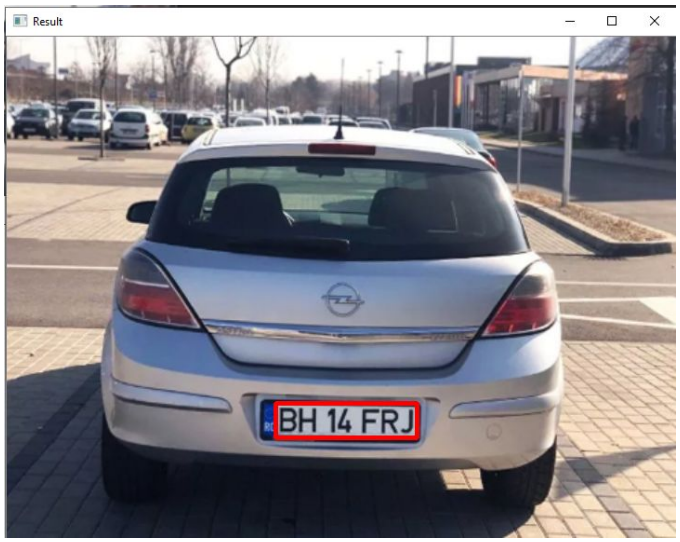
Imaginea intersectie dintre color si label:



Imaginea binara a tablitei:

Ultima prelucrare a tablitei:

Imaginea rezultat:



5. Concluzii

- Gradul de implinire a obiectivelor propuse

La inceputul laboratorului de proiect, mi-am propus sa extrag caracterele de pe fiecare placuta de inmatriculare. Pe parcurs, am realizat ca este dificil sa compar fiecare caracter etichetat in imagine binara cu literele si cifrele de pe placuta, asa ca am ramas la varianta in care extrag doar zona cu placuta, pe care o si incadrez intr-un dreptunghi rosu peste imaginea originala.

- Observatii asupra rezultatelor obtinute

Sansele ca rezultatul obtinut sa fie gasit sunt mult mai mari daca imaginea nu contine obiecte care ar putea sa fie interpretate ca placute de inmatriculare. De asemenea, pozele cu masini ar fi indicat sa aiba placuta putin inclinata spre dreapta

- Directii viitoare de dezvoltare

In primul rand, cel mai bun exemplu de dezvoltare ulterioara este chiar extragerea caracterelor din imaginile rezultate in acest proiect. De asemenea, o alta idee de dezvoltare ar putea fi extragerea imaginilor dintr-un video si mai apoi, detectarea automata a placutei de inmatriculare. Cea din urma varianta de dezvoltare ulterioara ar fi implementarea codului pe o placa Raspberry Pi sau Arduino si impreuna cu o camera, sa se plimbe intr-o parcare si sa detecteze placutele de inmatriculare ale masinilor parcate.

Bibliografie

- Suportul de curs si indrumatoarele de laborator
- openCV API
- https://en.wikipedia.org/wiki/Automatic_number-plate_recognition
- https://en.wikipedia.org/wiki/Edge_detection