Segmentarea participanțiilor la trafic

Oprica Andrei

Informatică Aplicată

Anul 3

Proiectului îi spune segmentarea participanțiilor la trafic, însă am ajuns doar la performanța de a detecta mașiniile ce circulă într-o anumită zonă de interes ce este selectată în argumentele funcției ce se ocupă cu procesarea și afișarea. Și camera ce filmează trebuie să filmeze dintr-un punct fix. Pentru videoclipuriile ce le am testat programul a detectat toate vehiculele din zona de interes, însă am avut câteva erori, acestea constând în detectarea nu foarte corectă a vehiculelor (adică detecta, dar putea găsi vehiculul și separat îi găsea și parbrizul, acesta ar fi un exemplu). Din această cauză nu pot nici să fac tracking pe vehicule, adică am reușit să încerc să fac tracking, dar pentru că programul detectează în unele cazuri și mașina și separat și parbrizul acesteia, astfel programul crede că aceași mașină sunt altele diferite.

În realizarea acestui proiect am utilizat python și am utilizat librăria cv2 (OpenCV).

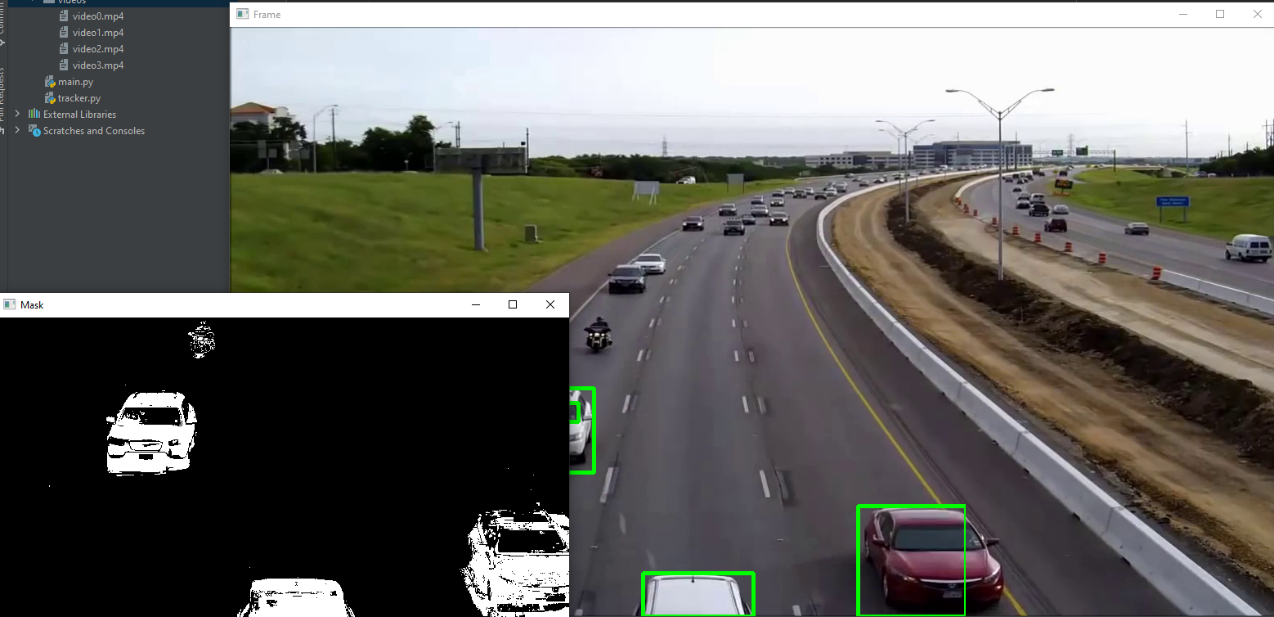
Partea de funcționalitate a proiectului constă în două fișiere .py:

Primul este un fișier denumit tracker.py în care am implementat o clasă denumită EuchideanDistTracker ce reține un dicționar în care v-or fi liste cu punctele participanțiilor și un numărător care indică identificatorul participantului. Această clasă conține și o metodă update() ce calculează distanța dintre punctele a două liste și în funcție de asta știe dacă este același participant sau este altul.

Al doilea fișier este un simplu fișier main.py în care import librăria cv2 și fișierul tracker. Apoi folosesc o funție ce primește cinci argumente: un string cu calea videoclipului ce trebuie analizat și patru int-uri reprezentând regiunea de interes pe care o analizăm. În funcție creez un obiect de tipul EuclideanDistTracker, un obiect ce extrage fundalul pentru detectarea participanțiilor. Apoi deschid videoclipul și îl iau pe cadre, din fiecare cadru îmi selectez regiunea de interes pentru a avea o detectie cât mai corectă. Asupra regiunii de interes aplic detectarea obiectelor, urmat de două threshold-uri (primul pentru a elimina umbrele create de obiecte și a doua pentru a obține o definire mai bună, prin eliminarea reflexiilor), urmat de un filtru bilateral în încercarea de a reduce reflexiile. După am căutat contururile cu ajutorul unei funcții oferite de cv2 și am căutat în acestea, iar dacă aveau o arie destul de mare le adaug la detecții. Până aici am facut detecție și a mers bine. În continuare am încercat să urmăresc participanții adăugând un identificator fiecărei detecții. Aici am avut probleme din cauza reflexiilor și a detrecției întrerupe a participanțiilor astfel rezulta că aceluiași participant i se schimba numarul de identificare. Prin urmare am scos partea de numerotare a participanțiilor. După sunt afișate cadrele pe rând. Aceată funcție este apelată din main cu cele 5 argumente și poate fi întreruptă prin apăsarea tastei escape.

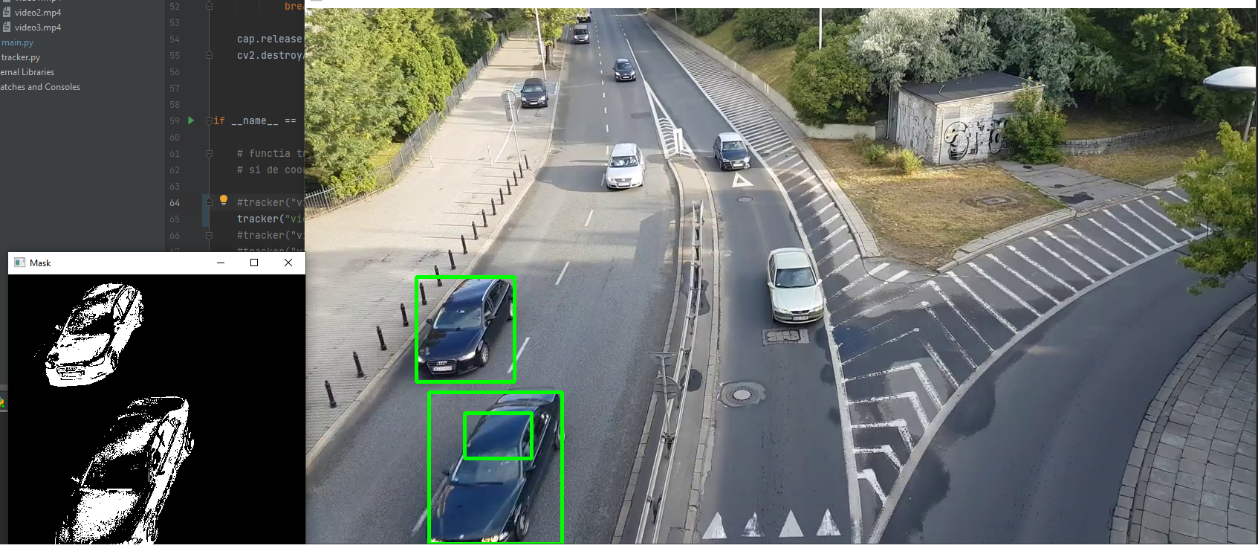
În cele ce urmează voi prezenta câteva rezultate a programului.

Primul exemplu:



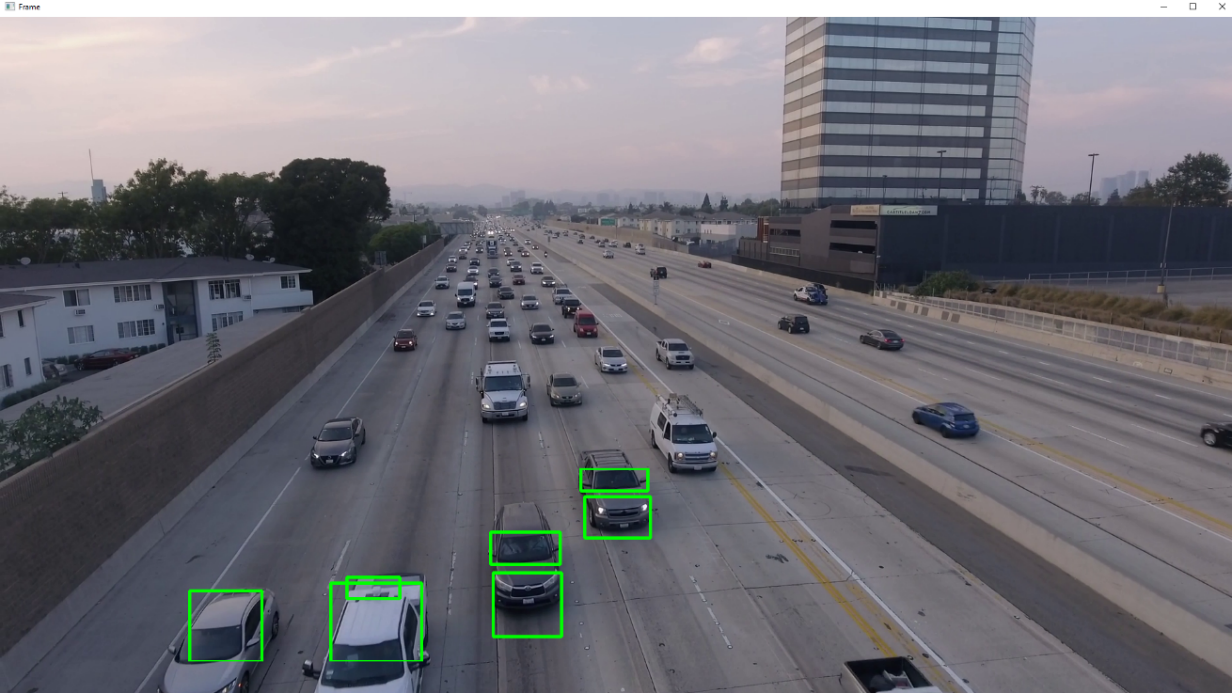
Partea alb-negru este masca pe care au fost aplicate threshold-urile și filtrul. De asemenea este zona de interes pe care se caută detecția. După cum se poate observa programul a detectat mașina albă din stânga, dar a detectat separat și parbrizul acesteia, ceea ce este puțin eronat.

Al doilea exemplu:



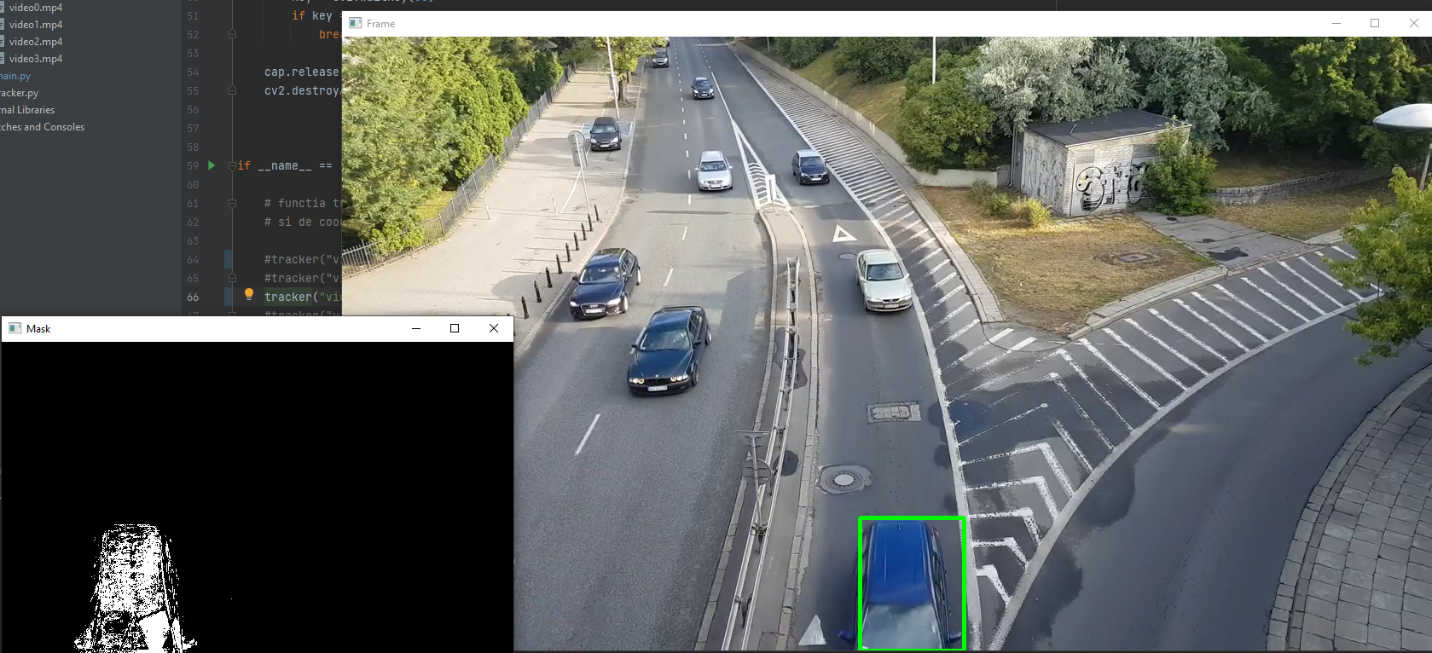
Și în cazul acestui exemplu se poate obseva, chiar mai bine, eroarea de la primul exemplu, acesta fiind detectarea parbrizului mașinii din dreapta în cazul acestui videoclip.

Al treilea exemplu:



În acest exemplu se poate observa mai bine fiind mai mulți participanți.

Al patrulea exemplu:



Acest exemplu este tot videoclipul de la exemplul al doilea, dar cu o altă regiune de interes. În acest exemplu programul a detectat bine.

Implementarea programului poate fi găsită la adresa: [https://github.com/AndreiOprica/appCV.](https://github.com/AndreiOprica/appCV)

În primă fază am urmărit un tutorial pentru a crea clasa EuclideanDistTracker și pentru a putea citi și afișa videoclipul apoi am folosit câteva maști pentru a încerca să optimizez programul pentru datele mele de intrare.

Bibliografie

* Tutorialul urmarit:

<https://youtu.be/O3b8lVF93jU>

* Câteva dintre videoclipuriile pe care am adaptat programul:
* <https://youtu.be/MNn9qKG2UFI>
* <https://youtu.be/PJ5xXXcfuTc>
* <https://youtu.be/QuUxHIVUoaY>
* Documentație:
* <https://docs.opencv.org/4.x/d7/d4d/tutorial_py_thresholding.html>
* <https://docs.opencv.org/3.4/d4/d86/group__imgproc__filter.html>
* <https://docs.opencv.org/4.x/d4/d73/tutorial_py_contours_begin.html>