

OE4DOS 2020/2021 – treći domaći zadatak

U arhivi uz domaći zadatak dat je jedan dashcam video (**video_road.mp4**). Automobil se kreće u jednoj kolovoznoj traci po autoputu, gde je traka ograničena punom ili isprekidanom belom linijom, ili punom žutom linijom. Potrebno je detektovati linije koje razdvajaju kolovozne trake. Detektovane linije (po jedna za svaku razdelnu liniju) iscrtati kao duži preko potojećih frejmova videa. Rešavanje zadatka je potrebno podeliti u više celina:

- 1) [15] Napisati funkciju ***segment_lanes*** koja za ulaznu sliku (*img_in*) izdvaja što bolje segment slike u kojem se nalaze linije koje razdvajaju saobraćajne trake. Ideja je izvršiti segmentaciju slike kreiranjem maske gde su sa 0 označeni delovi slike koji nisu od interesa (nebo, trava i sl) a sa 1 delovi slike koji čine put i linije koje razdvajaju saobraćajne trake. Primenom ove mape na ulaznu sliku dobija se segmentirana slika na kojoj su ostali samo regioni od interesa- Nije neophodno da segmentacija bude idealna, već što je bolje moguća uz razumne performanse kako bi se što više olakšalo narednim koracima algoritma. **Segmentacija se vrši na osnovu karakteristika regiona i nije dozvoljeno ručno unošenje koordinata regiona od interesa.**

U izveštaju detaljno opisati način realizacije funkcije i prikazati međurezultate i finalne rezultate. Potrebno je da funkcija radi robusno. Funkciju razvijati na datim test slikama a robusnost segmentacije testirati primenom funkcije nad celom video sekvencom.

- 2) [15] Napisati funkciju ***canny_edge_detection*** koja za ulaznu sliku (*img_in*), standardnu devijaciju Gausove funkcije (*sigma*) i vrednosti nižeg (*threshold_low*) i višeg praga (*threshold_high*) detektuje ivice korišćenjem Kanijevog algoritma koji se sastoji iz sledećih koraka:

1. Filtriranje ulazne slike Gausovom funkcijom zadate standardne devijacije. Za radijus Gausove funkcije usvojiti vrednost od 3 sigma.
2. Određivanje horizontalnih i vertikalnih gradijenata tako filtrirane slike. Koristiti Sobelov operator za izdvajanje gradijenata.
3. Određivanje magnitude i ugla gradijenta.
4. Kvantizacija gradijenta na jedan od 4 pravaca -45° , 0° , 45° , 90° .
5. Potiskivanje vrednosti gradijenata koje ne predstavljaju lokalne maksimume.
6. Određivanje mapa jakih i slabih ivica na osnovu vrednosti nižeg i višeg praga.
7. Uključivanje u izlaznu mapu ivica onih slabih ivica koje su povezane sa nekom jakom ivicom.

U izveštaju detaljno opisati svaki od ovih navedenih koraka i prikazati međurezultate. Prilikom testiranja funkcije koristiti neku od zadatih test slika. Testirati funkciju sa različitim vrednostima ulaznih parametara i iskomentarisati rezultate.

- 3) [15] Napisati funkciju ***get_line_segments*** koja na osnovu matrice izdvojenih ivica i pravca linije zadate u normalnoj reprezentaciji izdvaja sve duži zadate minimalne dužine koje se nalaze na datom pravcu. Duž predstavlja skup ivičnih tačaka koje su susedne i sve se nalaze na zadatom pravcu. Svaka duž je određena početnom i krajnjom tačkom. Da bi duž bila validna potrebno je da ispunjava sledeće kriterijume:
 - Sve tačke duži između početne i krajnje se nalaze na zadatom pravcu
 - Dozvoljeno je ignorisati nedostajuće segmente maksimalne dužine koja se zadaje prilikom poziva funkcije
 - Smatra se da određeni piksel na zadatom pravcu pripada ivici ukoliko se u njegovoj okolini, čija veličina se zadaje kao parametar funkcije, nalazi ivični piksel

Ulazni parametri:

img_edges– slika sa detektovanim ivičnim pikselima sa koje se detektuju linije

line – niz od dva elementa (theta, rho) kojim se zadaje pravac linije
min_size – minimalna dužina segmenta (u pikselima) da bi duž bila detektovana, sve linije koje su manje od ove veličine treba da budu ignorisane. Povratni argument ove funkcije su sve duži čija je dužina veća od ovog parametra
max_gaps – maksimalne veličine prekida (u pikselima) koje se mogu ignorisati prilikom detekcije duži
tolerancy – radijus okoline u okviru koje se traže ivični pikseli

Izlazni parametri:

line_segments – niz detektovanih duži pri čemu je svaka duž predstavljena sa koordinatama početne i krajnje tačke duži

- 4) [15] Napisati funkciju **lane_detection** koja kao jedini ulazni argument prihvata sliku (jedan video frejm) i vraća dva niza od po 4 elementa $[xl1, yl1, xl2, yl2]$, $[xr1, yr1, xr2, yr2]$ koji predstavljaju koordinate tačaka koje jednoznačno određuju levu i desnu liniju trake u kojoj se automobil kreće.

U izveštaju je potrebno prikazati rezultate međukoraka algoritma za 6 karakterističnih frejmova videa, kao i konačno rešenje sa ucrtanim linijama koje vraća funkcija **lane_detection**. Obezbediti da glavni program kreira izlazni video sa ucrtanim linijama.

Hint: Primetiti da su kolovozne linije karakterističnih boja. Razmisliti o odabiru kolor sistema za lakše izdvajanje traženih boja. Primetiti uticaj pozadine (nebo, trava, drveće, beli automobil) na detekciju željenih linija. Razmisliti po čemu se linije kolovozne trake razlikuju od ostalih linija koje se detektuju na slici. Iskoristiti ove informacije prilikom podešavanja parametara hough funkcije ili nekog drugog dela vašeg algoritma. Pri konačnom prikazu linija kolovozne trake preporučljivo je uzeti da se linije uvek iscrtavaju od dna slike pa do neke visine slike, kako bi se zadržala uniformnost prikaza na videu. Ovo je moguće uraditi na osnovu izračunatog nagiba linije i određivanja x koordinata tačaka za koje smo uzeli predefinisane y koordinate.

[Bonus 10] Detektovati linije koje razdvajaju kolovozne trake na video sekvenci **video_bonus.mp4**

Rešenje domaćeg zadatka napisati u okviru jedne sveske **domaci3_gg_bbb.ipynb** pri čemu je rešenje posebnih tačaka potrebno podeliti u posebne ćelije (ili više ćelija za jednu tačku ako ima više smislenih celina).

Napomena: Nemojte slati slike koje su date uz zadatak. Skripta za testiranje treba da bude u okviru direktorijuma **domaci3_gg_bbb** pri čemu se podrazumeva da se ulazne sekvence nalaze na relativnoj putanji **../sekvence**.

Fajlove **domaci3_gg_bbb.ipynb**, **izveštaj (domaci3_gg_bbb.pdf)**, kao i sve dodatne fajlove potrebne za pokretanje glavnog programa zapakovati u **domaci3_gg_bbb.zip** i okačiti na OneDrive. Link ka rešenju poslati na adresu elmezeni@etf.rs i meja@etf.rs sa subjectom **OE4DOS treci domaci**.

Rok za predaju rešenja domaćeg zadatka je **petak 22.01.2021.**
Svaki dan kašnjenja povlači -10% osvojenih poena!