

OE4DOS 2020/2021 – treći domaći zadatak

1. U arhivi uz domaći zadatak dat je skup slika sata. Vreme pokazuju kazaljke za sate i minute koje su u crnoj boji. Na nekim satovima postoji i kazaljka za sekunde koja može a ne mora biti crne boje. Potrebno je odrediti vreme koje prikazuju satovi. Rešavanje zadatka je potrebno podeliti u više celina:

- 1) [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.

- 2) [10] 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 (θ , ρ) 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.

U izveštaju detaljno opisati korake u radu funkcije i prikazati međurezultate. Prilikom testiranja funkcije koristiti neku od zadatih test slika. Testirati funkciju sa različitim vrednostima ulaznih parametara i iskomentarisati rezultate. Rezultate prikazati vizuelno iscrtavanjem izdvojenih duži preko originalne slike.

- 3) [10] Potrebno je napisati funkciju ***extract_time*** koja kao jedini ulazni argument prihvata matricu slike sata i vraća tuple od dva elementa koji predstavljaju sate i minute koje pokazuje sat na datoj slici. Voditi računa o tome da python funkcije ***skimage.transform.hough_line*** i ***skimage.transform.hough_line_peaks*** određuju ugao u odnosu na horizontalnu osu.

Ulazni parametri:

image – ulazna slika. Može biti grayscale ili u boji, kao i različitih formata (.png, .jpg). Obezbediti proveru naziva slike uz pomoć biblioteke ***os*** (funkcije ***os.listdir(<folder path>)***).

Izlazni parametri:

(hour, minute) – tuple koji se sastoji od brojeva koji predstavljaju sate i minute.

Hint: Primetiti da je kazaljka koja pokazuje sekunde ili dosta tanja od ostalih kazaljki ili je crvene boje, što znači da će po prebacivanju u sivu sliku dati manji intenzitet gradijenta. Iskoristiti ove informacije prilikom podešavanja parametara detektora ivica ili nekog drugog dela vaše funkcije.

Bonus [2]: Napisati posebnu funkciju ***extract_time_bonus*** koja pored podataka o satima i minutima vraća i podatak o sekundama za slike na kojima kazaljka za sekunde postoji. Ukoliko kazaljka za sekunde nije prisutna vratiti podatak -1. Smatrati da ukoliko postoji kazaljka za sekunde nije preklapljena sa ostalim kazaljkama. Bonus zadatak se ne uzima u obzir ako osnovni zadatak nije potpuno rešen!

2. U arhivi uz domaći zadatak dat je skup slika novčića. Novčići su raspoređeni na karakterističnoj podlozi bez preklapanja. Postoje dve vrste novčića sa vrednostima 1 i 5 dinara. Broj novčića je različit na slikama, ali su na svakoj slici prisutni novčići obe vrste. Potrebno je detektovati i segmentisati novčiće, odrediti površine segmenata i na osnovu njih klasifikovati novčiće. Cilj je odrediti broj novčića po vrsti. Rešavanje zadatka je potrebno podeliti u više celina:

- 1) [5] Napisati funkciju ***coin_mask*** koja za ulaznu sliku (***img_in***) kreira binarnu masku takvu da vrednost 1 imaju pikseli koji pripadaju novčićima, dok vrednost 0 ima ostatak slike. Potrebno je da segmentacija bude što preciznija i da novčići nemaju rupice. **Segmentacija se vrši na osnovu karakteristika novčića ili pozadine, geometrije, i uz pomoć morfoloških operacija.**

U izveštaju detaljno opisati način realizacije funkcije i prikazati međurezultate i finalne rezultate. Potrebno je da funkcija radi robusno.

- 2) [10] Napisati funkciju ***bw_label*** koja na osnovu binarne slike izdvojenih regiona od interesa pravi listu povezanih regiona i dodeljuje im labelu. Pod povezanim regionom se smatra površina sa pikselima vrednosti 1 (iz binarne slike) koji su međusobno povezani¹. Algoritam za određivanje povezanih regiona obrađen je na predavanjima.

Ulazni parametri:

img_bin – binarizovana slika (sa izdvojenim novčićima)

Izlazni parametri:

bw_img – slika sa labeliranim nepreklapajućim elementima. Pikseli elemenata imaju vrednost jednaku labeli elementa. Pikseli pozadine imaju vrednost 0.

U izveštaju detaljno opisati korake u radu funkcije i prikazati međurezultate (najlakše je različitim labelama dodeliti različite boje). Prilikom testiranja funkcije koristiti neku od zadatih test slika

¹ povezanost se definiše 8-connect pravilom. Dva piksela su povezana ako se jedan nalazi u okolini 8 drugog, to jest ako se dodiruju stranicama ili temenima (dijagonalni pikseli).

ili kreirati jednostavniju. Dati tabelu pronađenog broja elemenata na svakoj od test slika u odnosu na pravi broj elemenata (novčića).

- 3) [10] Napisati funkciju ***coin_classification*** koja kao jedini ulazni argument prihvata sliku, a vraća niz od dva broja koji predstavljaju broj novčića sa slike koji pripadaju klasi manje površine (1 dinar) i klasi veće površine (5 dinara) respektivno, koristeći prethodno realizovane funkcije.

U izveštaju je potrebno prikazati rezultate međukoraka algoritma za 2 karakteristične slike. Na osnovu izlaza ove funkcije na originalnoj slici ispisati preko svake površine (novčića) kojoj klasi pripada i priložiti tabelu koja definiše koliko novčića koje klase ima na svakoj slici i koja je greška u odnosu na referentnu vrednost (data tabela u materijalima). Obezbediti da glavni program čuva slike sa obeleženim površinama i klasama novčića.

Hint: Primetiti da je površina na kojoj se nalaze novčići karakteristične nijanse i saturacije. Obezbediti da površine budu uniformnije pomoću neke od metoda filtriranja. U slučaju pojave rupica na binarizovanim površinama koristiti morfološke operacije za popravku, ali voditi računa da se ne izmeni značajno površina elementa u odnosu na površinu novčića sa originalne slike.

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).

Uspešno rešen zadatak može doneti najviše polovinu poena. Za pun broj poena potrebno je da kod bude iskomentaran, da rezultati budu smisleno predstavljeni, da za svaku tačku postoji diskusija o načinu na koji je rešavan zadatak, koji je bio način razmišljanja i kako su i koji rezultati dobijeni. Dakle nije dovoljno dobiti smilen rezultat već je potrebno kroz izveštaj pokazati da je taj rezultat posledica smislenog procesa i da vi razumete to što ste uradili. Finalna verzija izveštaja ne sme da sadrži nezakomentarisane *interact* komande. *Interact* se može koristiti prilikom određivanja parametara, ali u finalnoj verziji izveštaja treba da stoje jasno definisani parametri i obrazloženje kako su dobijeni. Za finalne vrednosti parametara potrebno je obrazložiti zašto je odabrana baš ta vrednost, koji su kriterijumi korišćeni i šta se dešava ako se upotrebi veća ili manja vrednost od odabrane.

Napomena: Nemojte slati slike koje su date uz zadatak. Skripta za testiranje, kao i svi fajlovi koji vam nisu bili zadati treba da budu u okviru direktorijuma ***domaci3_gg_bbb*** pri čemu se podrazumeva da se ulazne sekvence nalaze na relativnoj putanji ***../sekvence/clocks***, i ***../sekvence/coins***.

Na primer: `I = imread('../sekvence/clocks/clock8.jpg');`

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 **ponedeljak 03.01.2022.**

Svaki dan kašnjenja povlači -10% osvojenih poena!