

19M041DOS2 2022/2023 – drugi domaći zadatak

[25] ZADATAK 1 – Realizacija Lucas-Kanade metode optičkog toka

U okviru ovog zadatka potrebno je implementirati funkciju kojom se realizuje Lucas-Kanade estimacija optičkog toka. Funkciju nazvati **lucasKanade**. Argumenti funkcije su:

Ulazi:

img1 – ulazna siva slika trenutnog frejma.

img2 – ulazna siva slika narednog frejma.

keypoints – niz dimenzija (N,2) sa pozicijama ključnih tačaka.

winSize – veličina prozora oko ključne tačke koji se koristi za estimaciju.

errThr – prag za odbacivanje loše detektovanih tačaka.

Izlaz:

flowVectors – niz dimenzija (N,2) sa vrednostima komponenti pomeraja u svakoj ključnoj tački.

validPoints – niz dimenzija N boolean vrednosti koji određuje da li je odgovarajuća ključna tačka pronađena u narednom frejmu. Ključna tačka se smatra izgubljenom ukoliko je njena nova pozicija izvan okvira frejma ili ukoliko je srednja kvadratna razlika normalizovanog prozora 3x3 oko stare i nove pozicije tačke veća od praga *errThr*.

Prilikom testiranja funkcije, iskoristiti neku od gotovih funkcija *corner_peaks*, *corner_harris* iz skimage biblioteke ili *cornerHarris*, *goodFeaturesToTrack* iz OpenCV biblioteke.

Testirati implementiranu funkciju na sekvenci slika iz direktorijuma **Can**. Prilikom testiranja, detektovati ključne tačke samo na provoj slici sekvence. U izveštaju prikazati ovu sliku sa ključnim tačkama. Odrediti vektore optičkog toka, iscrtati strelice koje predstavljaju ove vektore na svakoj ključnoj tački i prikazati ovaj rezultat u izveštaju. Nakon toga, izvršiti praćenje ključnih tačaka kroz sve slike sekvence. Praćenje se obavlja tako što se na osnovu pozicije ključnih tačaka trenutnog frejma određuje pozicija ključnih tačaka narednog frejma. **Prate se samo validne ključne tačke**. Dakle, broj ključnih tačaka će se smanjivati iz frejma u frejm. U izveštaju prikazati pozicije ključnih tačaka na frejmovima 1, 5, 9 i 13. Prilikom estimacije optičkog toka koristiti prozor 5x5. Podesiti vrednost praga greške tako da se odbaci najveći broj loše detektovanih pomeraja. U okviru direktorijuma **CanLK** sačuvati slike svih frejmova sa obeleženim validnim ključnim tačkama.

[15] ZADATAK 2 – Realizacija iterativne Lucas-Kanade metode optičkog toka

Prethodno realizovana funkcija podrazumeva male pomeraje ključnih tačaka između frejmova. U slučaju većih pomeraja javlja se značajna greška u proceni optičkog toka. Zbog toga je potrebno realizovati poboljšanu verziju prethodne funkcije u kojoj se procena vektora pomeraja obavlja iterativno. Na početku se svi vektori pomeraja inicijalizuju na osnovu ulaza *initFlow* a u svakoj narednoj iteraciji se koristi prethodno procenjena vrednost. Prilikom računanja temporalnog gradijenta posmatra se razlika piksela na poziciji ključne tačke trenutnog frejma i piksela na poziciji određenoj vektorom pomeraja (iz prethodne iteracije) narednog frejma. Funkciju nazvati **iterLucasKanade**. Argumenti funkcije su:

Ulazi:

img1 – ulazna siva slika trenutnog frejma.

img2 – ulazna siva slika narednog frejma.

keypoints – niz dimenzija (N,2) sa pozicijama ključnih tačaka.

winSize – veličina prozora oko ključne tačke koji se koristi za estimaciju.

numIter – broj iteracija.

initFlow – niz dimenzija (N,2) sa inicijalnim vrednostima vektora pomeraja u svakoj ključnoj tački.

errThr – prag za odbacivanje loše detektovanih tačaka.

Izlaz:

flowVectors – niz dimenzija (N,2) sa vrednostima komponenti pomeraja u svakoj ključnoj tački.

validPoints – niz dimenzija N boolean vrednosti koji određuje da li je odgovarajuća ključna tačka pronađena u narednom frejmu. Ključna tačka se smatra izgubljenom ukoliko je njena nova pozicija izvan okvira frejma ili ukoliko je srednja kvadratna razlika normalizovanog prozora 3x3 oko stare i nove pozicije tačke veća od praga *errThr*.

Testirati implementiranu funkciju na sekvenci slika iz direktorijuma **Can**. Testiranje obaviti na način opisan u zadatku 1. Prilikom testiranja koristiti veličinu prozora 9x9 i broj iteracija 7. Smatrati da je inicijalna vrednost pomeraja jednaka 0. U okviru direktorijuma **CanLKiter** sačuvati slike svih frejmova sa obeleženim validnim ključnim tačkama.

[15] ZADATAK 3 – Realizacija piramidalne Lucas-Kanade metode optičkog toka

U slučaju velikih pomeraja može doći do temporalnog alijasinga i iterativna procena optičkog toka može iskonvergirati u lokalni minimum. Kako bi se omogućila detekcija velikih pomeraja ideja je da se iskoristi Gausova piramida. Na taj način se veliki pomeraji na višim nivoima piramide svode na manje pomeraje i moguće ih je proceniti korišćenjem iterativnog metoda. Funkciju nazvati **pyrLucasKanade**. Argumenti funkcije su:

Ulazi:

img1 – ulazna siva slika trenutnog frejma.

img2 – ulazna siva slika narednog frejma.

keypoints – niz dimenzija (N,2) sa pozicijama ključnih tačaka.

winSize – veličina prozora oko ključne tačke koji se koristi za estimaciju.

numIter – broj iteracija.

levelNum – broj nivoa Gausove piramide.

levelScale – faktor skaliranja između susednih nivoa piramide (npr ima vrednost 2 ukoliko se radi decimacija 2 puta po obe dimenzije).

errThr – prag za odbacivanje loše detektovanih tačaka.

Izlaz:

flowVectors – niz dimenzija (N,2) sa vrednostima komponenti pomeraja u svakoj ključnoj tački.

validPoints – niz dimenzija N boolean vrednosti koji određuje da li je odgovarajuća ključna tačka pronađena u narednom frejmu. Ključna tačka se smatra izgubljenom ukoliko je njena nova pozicija izvan okvira frejma ili ukoliko je srednja kvadratna razlika normalizovanog prozora 3x3 oko stare i nove pozicije tačke veća od praga *errThr*.

Prilikom implementacije funkcije za kreiranje Gausove piramide može se iskoristiti funkcija *pyramid_gaussian* iz *skimage.transform* biblioteke. Voditi računa da se pri prelasku na novi nivo piramide skaliraju i pozicije ključnih tačaka.

Testirati implementiranu funkciju na sekvenci slika iz direktorijuma **Can**. Testiranje obaviti na način opisan u zadatku 1. Prilikom testiranja koristiti veličinu prozora 9x9, broj iteracija 7, ukupno 3 nivoa Gausove piramide, sa faktorom decimacije 2 između nivoa. U okviru direktorijuma **CanLKpyr** sačuvati slike svih frejmova sa obeleženim validnim ključnim tačkama.

[20] ZADATAK 4 – Praćenje objekta

U okviru ovog zadatka potrebno je realizovati praćenje lica čoveka na sekvenci slika. Slike se nalaze u okviru direktorijuma **Man**. Potrebno je na prvoj slici iz sekvence ručno definisati pravougaonik koji obuhvata lice čoveka. Pravougaonik se zadaje preko koordinata gornjeg levog ugla, visine i širine. Tokom postupka praćenja menja se samo koordinata gornjeg levog ugla dok su dimenzije pravougaonika konstantne. Praćenje se obavlja tako što se na početku detektuju ključne tačke u okviru definisanog pravougaonika. Nakon toga se koriste algoritmi optičkog toga projektovani u prethodnim tačkama za određivanje pomeraja ključnih tačaka od frejma do frejma. Pomeraj pravougaonika oko lica predstavlja srednju vrednost pomeraja svih validnih ključnih tačaka. Nakon određivanja nove pozicije pravougaonika odbacuju se sve one ključne tačke koje se nalaze van njegovih granica. Kako se na ovaj način broj ključnih tačaka smanjuje s vremenom, potrebno je na svakih 20 frejmova ponovo detektovati ključne tačke u okviru pravougaonika od interesa.

Potrebno je sačuvati dve izlazne video sekvence. Jedna video sekvenca (*man_track.mp4*) predstavlja originalni video sa iscertanim pravougaonikom oko lica na procenjenim pozicijama. Druga video sekvenca ima ucrtane i validne karakteristične tačke (*man_track_key.mp4*).

[15] BONUS ZADATAK – Stabilizacija videa

Snimiti ili preuzeti neku video sekvencu u kojoj jasno postoje globalni pokreti kamere. Dobar primer je snimanje videa iz ruke u toku hodanja ukoliko uređaj kojim se snima nema uključenu stabilizaciju. Pre obrade izvršiti skaliranje frejmova kako bi se ubrzalo procesiranje. U toku procesiranja potrebno je odrediti i globalno kompenzovati dominantne pomeraje u video sekvenci. Dominantni pomeraji nisu najveći pomeraji već oni koji se konzistentno javljaju na najvećem broju piksela. U izveštaju detaljno opisati primenjeno rešenje i prikazati međurezultate procesiranja. Stabilizovanu video sekvencu snimiti u poseban video fajl.

NAPOMENA ZA SVE TAČKE:

U izveštaju detaljno analizirati problem, opisati plan i postupak rešavanja uz argumentovanje svakog koraka i demonstriranje slikama međurezultata.

Rešenje za sve tačke ovog domaćeg napisati u okviru jedne sveske *domaci1_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 iskomentarisan, 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 smislen 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 nezakomentrisane *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 *domaci2_gg_bbb* pri čemu se podrazumeva da se ulazne sekvence nalaze na relativnoj putanji *../imgs*.
Na primer: `I = imread('../imgs/book/book_model.jpg');`

Fajlove *domaci2_gg_bbb.ipynb*, korišćene ulazne slike koje nisu bile u zadatku (ako ih ima), **izveštaj** (*domaci2_gg_bbb.pdf*), kao i sve dodatne fajlove potrebne za pokretanje glavnog programa zapakovati u *domaci2_gg_bbb.zip* i okačiti na OneDrive. Link ka rešenju poslati na adresu elmezeni@etf.rs sa subjectom **MS1DOS2 drugi domaci**.

Rok za predaju rešenja domaćeg zadatka je **nedelja 28.05.2023.**
Svaki dan kašnjenja povlači -10% osvojenih poena!