

## 19M041DOS2 2022/2023 – prvi domaći zadatak

### [15] ZADATAK 1 – Realizacija DoG detektora obeležja

U okviru ovog zadatka potrebno je realizovati detektor karakterističnih tačaka koje su stabilne u odnosu na translaciju, rotaciju i skaliranje. Kako bi se ostvarila invarijantnost u odnosu na skaliranje karakteristične tačke se traže u okviru DoG (Difference of Gaussian) piramide, čime se osim prostorne pozicije određuje i karakteristična skala obeležja. Funkciju je potrebno realizovati u više koraka navedenih u nastavku.

- 1) [5] Prvi korak predstavlja realizaciju funkcije kojom se kreiraju nedecimirana Gausova piramida i DoG piramida ulazne slike. Funkciju nazvati **createPyramid**. Argumenti funkcije su:

#### Ulazi:

**img** – ulazna slika sadrži jednu ravan a vrednosti piksela su u opsegu (0,1)  
**sigma0** – standardna devijacija Gausovog filtra za određivanje nultog nivoa piramide  
**k** – faktor povećanja standardne devijacije filtra za određivanje narednog nivoa piramide  
**numLevels** – DoG piramida sadrži sve nivoe od 0 do numLevels-1

#### Izlaz:

**GaussianPyramid** – lista u okviru koje se nalaze nivoi kreirane Gausove piramide.  
Ova piramida sadrži sve nivoe od 0 do numLevels-1.

**DoGPyramid** – lista u okviru koje se nalaze razlike susednih nivoa Gausove piramide.  
L-ti nivo DoG piramide predstavlja razliku L-1 i L-tog nivoa Gausove piramide.

Obratiti pažnju da je za kreiranje nivoa DoG piramide od 0 do numLevels-1 potrebno kreirati Gausovu piramidu sa nivoima od -1 do numLevels-1.

Standardna devijacija Gausovog filtra koji se koristi za dobijanje  $l$ -tog nivoa Gausove piramide je:

$$\sigma_l = k^l \sigma_0$$



Slika 1. Primer izgleda Gausove i DoG piramide

U izveštaju opisati realizaciju i prikazati rezultate funkcije. Prilikom testiranja koristiti sledeće parametre:  $\sigma_0 = 1$ ,  $k = \sqrt{2}$ ,  $N_L = 5$ .

- 2) [10] Nakon toga potrebno je pronaći tačke koje predstavljaju lokalne ekstremume u prostoru i skali DoG piramide. Potrebno je da funkcija ima mogućnost odbacivanja odziva koji pripadaju uniformnim površinama (mala apsolutna vrednost DoG odziva) ili pripadaju ivicama (nije zadovoljena kriterijumska funkcija na bazi Hesijana). Funkciju nazvati **detectDoGFeatures**. Argumenti funkcije su:

**Ulazi:**

**DoGPyramid** – lista u okviru koje se nalaze razlike susednih nivoa Gausove piramide.

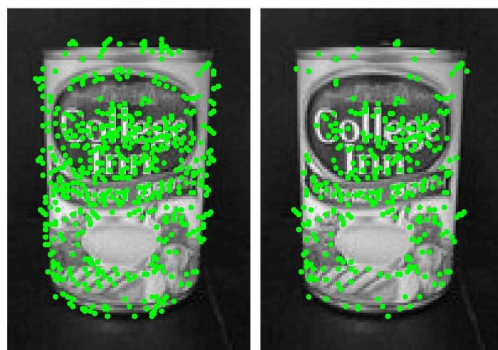
L-ti nivo DoG piramide predstavlja razliku L-1 i L-tog nivoa Gausove piramide.

**uniformThr** – vrednost praga kojom se odbacuju uniformne površine

**edgeThr** – vrednost praga kojom se odbacuju ivični odzivi

**Izlaz:**

**featureLocations** – Nx3 matrica koja sadrži N detektovanih lokacija obeležja koja zadovoljavaju postavljene kriterijume. Svako detektovano obeležje predstavlja uređenu trojku ( $x, y, level$ ).



Slika 2. Primer detektovanih karakterističnih tačaka bez (levo) i sa (desno) potiskivanjem ivičnih odziva

U izveštaju opisati realizaciju i prikazati rezultate funkcije. Prilikom testiranja koristiti sledeće parametre:  $\theta_u = 0.03$ ,  $\theta_e = 12$  - za pragove detektora.

## [15] ZADATAK 2 – Realizacija BRIEF deskriptora obeležja

U okviru drugog zadatka potrebno je opisati detektovana obeležja korišćenjem BRIEF deskriptora. Ovaj deskriptor opisuje region oko karakteristične tačke tako što se sprovodi niz testova poređenja prethodno odabranih parova tačaka. Rezultat svakog poređenja predstavlja jedan bit izlaznog deskriptora. Funkciju je potrebno realizovati u više koraka navedenih u nastavku.

- 1) [5] Pre kreiranja BRIEF deskriptora potrebno je generisati skup parova tačaka koji će se koristiti prilikom poređenja. Zbog toga je u okviru ove tačke potrebno realizovati funkciju **generateBRIEFpattern** koja treba da za lokalno susedstvo određene veličine i za zahtevani broj testova generiše isto toliko parova koordinata tačaka. Smatrati da se ove koordinate određuju na slučajan način po uniformnoj raspodeli. Argumenti funkcije su:

**Ulazi:**

**patchSize** – dimenzije lokalnog susedstva, smatrati da je susedstvo kvadratno

**testNum** – broj testova poređenja, odnosno broj parova koje je potrebno izgenerisati

**Izlaz:**

**p1Coords** – lista od testNum koordinata tačaka koje se koriste kao prve u poređenju.

**p2Coords** – lista od testNum koordinata tačaka koje se koriste kao druge u poređenju.

Kako bi obezbedili ponovljivost deskriptora koristiti fiksiranu vrednost *seed*-a. Vrednost *seed*-a treba da bude različita prilikom kreiranja slučajnih koordinata za prvu i drugu tačku.

U izveštaju prikazati generisani obrazac poređenja za prozor dimenzija 9x9 i 256 testova gde su parovi tačaka koje učestvuju u testovima povezani linijama.

- 2) [10] Napisati funkciju **computePyrBRIEF** koja za određene lokacije obeležja u prostoru Gausove piramide izračunava BRIEF deskriptore. Argumenti funkcije su:

**Ulazi:**

**GaussianPyramid** – Gausova iz koje se izračunava BRIEF opis obeležja

**featureLocations** – lokacije obeležja u okviru Gausove piramide

**k** – faktor povećanja standardne devijacije filtra za određivanje narednog nivoa piramide

**patchSize** – dimenzije lokalnog susedstva na nivou 0

**p1Coords** – lista koordinata tačaka koje se koriste kao prve u poređenju.

**p2Coords** – lista koordinata tačaka koje se koriste kao druge u poređenju.

**Izlaz:**

**descBRIEF** – matrica boolean vrednosti dimenzija KxN pri čemu je N broj deskriptora koji je izračunat, a K predstavlja broj testova.

**locBRIEF** – matrica 3xN koja sadrži informacije o poziciji deskriptora u prostoru i skali.

BRIEF deskriptor se na nivou 0 određuje na prozoru veličine patchSize. Za više nivoe potrebno je srazmerno skalirati veličinu prozora za faktor  $\sigma_i / \sigma_0$ . Voditi računa da je potrebno skalirati i pozicije test tačaka. Ukoliko se neko obeležje nalazi na ivicama slike tako da nema kompletno lokalno susedstvo, za takvo obeležje se ne računa deskriptor.

### [15] ZADATAK 3 – Realizacija funkcije za uparivanje obeležja

U okviru ovog zadatka potrebno je napisati funkciju za uparivanje obeležja **matchBRIEF**. Funkcija prihvata 4 argumenta sa deskriptorima i lokacijama obeležja iz dve slike. Uparivanje se obavlja određivanjem Hamingovog rastojanja između vrednosti deskriptora. Uparivanje se obavlja tako što se za svako obeležje iz jednog skupa odrede dva najbliža obeležja iz drugog skupa. Par se prihvata ukoliko je količnik rastojanja najboljeg i drugog najboljeg para manji od zadatog praga. Potrebno je obezbediti mogućnost i unakrsnog testiranja parova (par se prihvata ukoliko zadovoljava količnički test i kada slike zamene mesta). Argumenti funkcije su:

**Ulazi:**

**descriptor1** – opisi obeležja prve slike

**location1** – lokacije obeležja prve slike

**descriptor2** – opisi obeležja druge slike

**location2** – lokacije obeležja druge slike

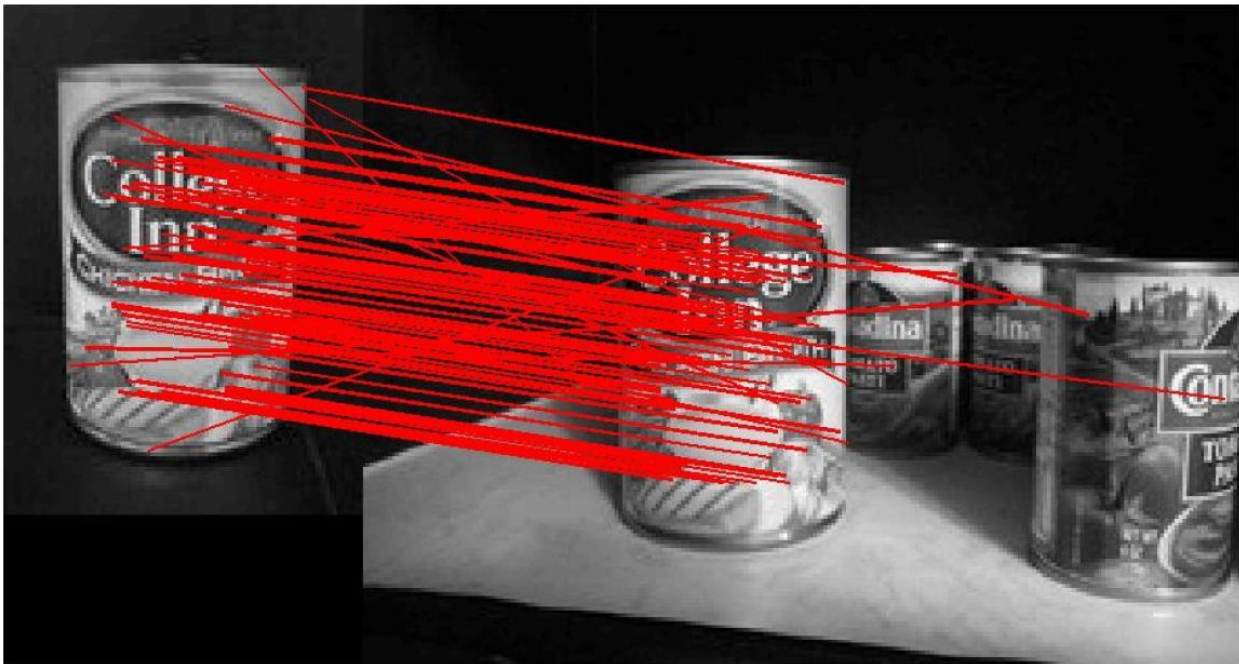
**ratioThr** – prag količničkog testa

**crossCheck** – False – ne koristi se unakrsno poređenje, True – koristi se unakrsno poređenje

**Izlaz:**

**matches** – lista od N uparenih tačaka. Svako polje liste sadrži listu sa lokacijom obeležja u prvoj i drugoj slici kao i distancu između obeležja.

U izveštaju opisati realizaciju i prikazati rezultate uparivanja slike **can\_model.jpg** sa ostalim slikama iz **imgs/can** direktorijuma. Prilikom testiranja koristiti  $\theta_r = 0.8$  - za prag količničkog testa.



Slika 3. Rezultat uparivanja BRIEF obeležja

#### [10] ZADATAK 4 – Testiranje robusnosti BRIEF deskriptora

U okviru ovog zadatka potrebno je istestirati robusnost razvijenog BRIEF deskriptora.

Postupak testiranja:

- 1) Prilikom testiranja koristiti sliku **can\_template.jpg**.
- 2) Detektovati karakteristična obeležja korišćenjem **detectDoGFeatures** funkcije.
- 3) Generisati geometrijsku transformaciju i primeniti je na sliku.
- 4) Detektovati karakteristična obeležja na transformisanoj slici korišćenjem **detectDoGFeatures** funkcije
- 5) Generisati BRIEF deskriptore korišćenjem funkcije **computePyrBRIEF** na polaznoj i transformisanoj slici.
- 6) Upariti deskriptore korišćenjem **matchBRIEF** funkcije. Prikazati ove parove kao na Slici 3.
- 7) Za svaki par odrediti da li je ispravno detektovan ili ne određivanjem prave lokacije polazne karakteristične tačke u transformisanoj slici korišćenjem poznate transformacione funkcije. Smatrati da je uparivanje ispravno ako pozicija detektovanog obeležja na transformisanoj slici odstupa od prave vrednosti za najviše  $\pm 3$  piksela.
- 8) Za svaki test slučaj odrediti ukupan broj detektovanih parova, ukupan broj ispravno detektovanih parova i njihov odnos.

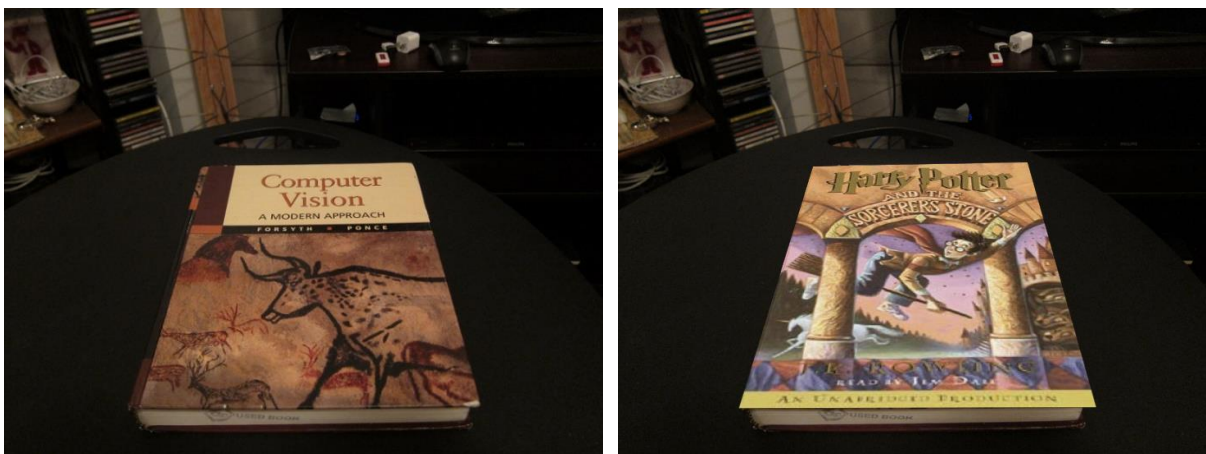
Primeniti prethodne korake za sledeće geometrijske transformacije: rotacija za 5, 10, 20, 40, 80, 120 stepeni, skaliranje 0.5, 2, 4 i proizvoljno odabrana afina transformacija i homografija. U izveštaju navesti rezultate za sve test slučajeve. Prikazati rezultate uparivanja iscrtavanjem linija koje povezuju uparene tačke na test slikama. Prokomentarisati dobijene rezultate u izveštaju.

## [20] ZADATAK 5 – Određivanje nepoznate homografije

U okviru ovog zadatka je potrebno korišćenjem prethodno razvijenih alata za pronalaženje parova tačaka odrediti nepoznatu transformaciju između dve slike. U direktorijumu **imgs/book** nalaze se slike iste knjige slikane iz različitih uglova. Model knjige dat je u slici **book\_model.jpg**. Nepoznata homografija se može odrediti pronalaženjem parova tačaka između slike modela i neke od ostalih slika knjige. Nakon određivanja parova tačaka može se iskoristiti neka od OpenCV funkcija za estimaciju nepoznate homografije.

Za svaku od test slika najpre pokušati estimaciju korišćenjem razvijenog BRIEF deskriptora. Ukoliko je ovaj postupak neuspešan iskoristiti SIFT deskriptor iz OpenCV-a. Za svaki test slučaj prokomentarisati dobijene rezultate.

Nakon procene nepoznate transformacije, moguće je zameniti prepoznati objekat nekim drugim objektom. U direktorijumu **imgs/book** nalazi se slika **hp\_cover.jpg**. Potrebno je omogućiti da se na svim test slikama umesto naslovne strane Computer Vision pojavi naslovna strana Harry Potter knjige. Rezultat pomenute operacije prikazan je na slici 4. Postupak zamene je potrebno obaviti automatski bez očitavanja koordinata sa zadatih slika.



Slika 4. Korišćenje homografije za zamenu planarnih objekata u sceni

U izveštaju detaljno opisati primenjeni postupak (sa međurezultatima uparivanja) i prikazati postignute rezultate. Prokomentarisati uspešnost BRIEF deskriptora za svaki od slučajeva.

## [10] BONUS 1 – Poboljšanje BRIEF-a

Predložiti i implementirati poboljšanje realizacije BRIEF deskriptora tako da se omogući rešavanje i slučajeva u kojima je inicijalno bio neuspešan. U izveštaju naglasiti koji problemi su rešavani, detaljno opisati način rešavanja i prikazati dobijene rezultate.

## 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 iskomentarisano, 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 **domaci1\_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 **domaci1\_gg\_bbb.ipynb**, korišćene ulazne slike koje nisu bile u zadatku (ako ih ima), **izveštaj (domaci1\_gg\_bbb.pdf)**, kao i sve dodatne fajlove potrebne za pokretanje glavnog programa zapakovati u **domaci1\_gg\_bbb.zip** i okačiti na OneDrive. Link ka rešenju poslati na adresu [elmezeni@etf.rs](mailto:elmezeni@etf.rs) sa subjectom **MS1DOS2 prvi domaci**.

Rok za predaju rešenja domaćeg zadatka je **nedelja 30.04.2023.**

**Svaki dan kašnjenja povlači -10% osvojenih poena!**