

UNIVERZITET U SARAJEVU
ELEKTROTEHNIČKI FAKULTET

PREPOZNAVANJE OBLIKA I OBRADA SLIKE

Projektni Zadatak br. 3

Studenti:

MUFTIĆ *Belma*, 1423/17260

LEMEŠ *Lamija*, 1474/17070

KRUPALIJA *Ehlimana*, 1431/17461

Odgovorni asistent:

MoE SUMEJJA PORČA

Decembar, 2018

Sadržaj

1	Prepoznavanje oblika	1
1.1	Koraci prije detekcije	1
1.2	Detekcija	2
1.3	Korištenje <i>slider</i> -a	3

1 Prepoznavanje oblika

1.1 Koraci prije detekcije

Detekcija oblika ćelije implementirana je u okviru *file*-a `main.py` koji se nalazi u folderu *Code*. Ta funkcija kao parametar prima *folder* u kojem se nalaze slike koje je potrebno detektovati, kao i najveći broj u imenu slika koje se nalaze u njemu.

Nakon što se sa *pickle* učitava prethodno kreirani model, potrebno je obraditi slike tako da se deskriptor može kreirati. Slike se nalaze u folderu *Validacija*, te kako je *dataset* inicijalno imao 90 slika, 10 slika je spadalo u validaciju. Kako su se nasumice birale slike, potrebno je kod staviti u `try-except` blok, u slučaju da se nađe na sliku koja se ne nalazi u *folderu*. Na osnovu karakteristike da je posmatrana ćelija tamnije obojena u odnosu na ostale ćelije, koristi se `inRange` da se kreira maska, nakon čega se primjeni morfologija da se uklone regije koje su slučajno našle u zadatom rasponu. Tako kreirana maska se šalje kao parametar u funkciju `HuMoments` da se kreira deskriptor, nakon čega se sa modelom testira kojoj klasi ćelija pripada.

Navedene funkcionalnosti implementirane su na sljedeći način:

```
B = cv2.inRange(Slika, lowerR, upperR)
shape = cv2.getStructuringElement
        (cv2.MORPH_ELLIPSE, (2,2))
nm = cv2.morphologyEx
        (B,cv2.MORPH_OPEN, shape)
shape2 = cv2.getStructuringElement
        (cv2.MORPH_ELLIPSE, (3,5))
NewMask= cv2.dilate(nm,shape2)
B=NewMask
huMoments = cv2.HuMoments
        (cv2.moments(B)).flatten()
for i in range(0,7):
    huMoments[i] = -1
    * m.copysign(1.0, huMoments[i])
    * m.log10(abs(huMoments[i]))
huMoments = huMoments.reshape(1,7)
rez = model.predict(huMoments)
```

1.2 Detekcija

Da bi se izvršila detekcija, potrebno je pronaći sve konture prethodno kreirane binarne slike, nakon čega se konture sortiraju po veličini, da bi se na prvom mjestu našla ona kontura koja opisuje ćeliju. Uz pomoć `boundingRect` moguće je pronaći kvadrat koji se nalazi oko zadane konture, te se isti nacрта. Rezultat testiranja modela se uz pomoć `putText` napiše u gornjem lijevom uglu slike, te se tako modifikovana slika prikaže korisniku.

Na sljedećim slikama prikazan je rezultat pokretanja detekcije nad nekoliko slika:



(a) Detektovana ćelija: *Lymphocyte*



(b) Detektovana ćelija: *Lymphocyte*



(c) Detektovana ćelija: *Neutrophil*



(d) Detektovana ćelija: *Something else*

Slika 1: Rezultati detekcije nad slikama za validaciju

1.3 Korištenje *slider*-a

Slider je objekat koji izvršava podjelu slike na isječke koji se zatim pojedinačno testiraju koristeći prethodno opisane koncepte. Ovaj objekat ima sljedeće atribute:

- **windowSize**: veličina prozora, odnosno kvadrata koji predstavlja isječak slike;
- **step**: korak, odnosno veličina koja određuje za koliko će se prozor pomjeriti (jednak i za visinu, i za širinu);
- **scaleChange**: veličina koja određuje koliko će se puta povećati prozor nakon što se dođe do kraja slike (ukoliko skaliranje bude neophodno);
- **imageSize**: veličina slike, koja određuje do kojih koordinata će se kretati vrijednosti koje određuju prozor;
- **coordinates**: koordinate vrhova kvadrata, neophodne za kreiranje isječaka slike;
- **maxScale**: maksimalni broj puta povećanja prozora (maksimalno skaliranje);
- **currentScale**: broj puta povećavanja prozora (broj skaliranja);
- **willScale**: veličina koja određuje da li će se skaliranje vršiti, te koja je postavljena na **True**, osim u slučaju da se objekat detektuje na slici.

Funkcija *slide* vrši pomijeranje prozora, dok funkcija *oneSlide* vrši pravljenje isječaka slike. Nakon što se prozor pomjeri i isječak kreira, vrši se detekcija na način kako je to prethodno opisano.

Ukoliko se ni u jednom isječku ne izvrši pozitivna detekcija, vrši se skaliranje, te se cijeli postupak zatim ponavlja. Ukoliko se ne detektuje nijedan objekat na slici, jednostavno se nastavlja s detekcijom na sljedećoj slici. Ukoliko dođe do detekcije, na slici se prikazuje pravougaonik koji se sastoji od svih isječaka na kojima je izvršena detekcija. Budući da je moguće da se detektuje više različitih vrsta oblika na slici (ukoliko se objekat nađe u više isječaka), konačni rezultat predstavlja ona klasa za koju je detektovano najviše isječaka.

Na sljedećim slikama prikazan je rezultat pokretanja detekcije nad nekoliko slika:



(a) Detektovana ćelija: *Lymphocyte*



(b) Detektovana ćelija: *Lymphocyte*



(c) Detektovana ćelija: *Neutrophil*



(d) Detektovana ćelija: *Something else*

Slika 2: Rezultati detekcije nad slikama za validaciju uz korištenje *slider*-a