

Ekstrakcija palete boja iz slika

Kvesić, Iva

Undergraduate thesis / Završni rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Electrical Engineering and Computing / Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:168:441988>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2025-03-15**



Repository / Repozitorij:

[FER Repository - University of Zagreb Faculty of Electrical Engineering and Computing repository](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1473

EKSTRAKCIJA PALETE BOJA IZ SLIKA

Iva Kvesić

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1473

EKSTRAKCIJA PALETE BOJA IZ SLIKA

Iva Kvesić

Zagreb, lipanj 2024.

ZAVRŠNI ZADATAK br. 1473

Pristupnica: **Iva Kvesić (0036534587)**
Studij: Elektrotehnika i informacijska tehnologija i Računarstvo
Modul: Računarstvo
Mentorica: prof. dr. sc. Sonja Grgić

Zadatak: **Ekstrakcija palete boja iz slika**

Opis zadatka:

Paleta boja igra ključnu ulogu u slikarstvu, dizajnu i fotografiji jer predstavlja alat umjetničkog izražavanja i utječe na percepciju i dojam slike. Pravilno odabrane boje mogu stvoriti ugodan i privlačan dojam, kreirati određenu atmosferu, doprinijeti prepoznatljivosti slike te usmjeriti pažnju gledatelja na određeni dio slike. Određivanje palete boja važno je i postupcima digitalne obrade slike ako što je kolorizacija, dekompozicija slike, segmentacija, klasifikacija i slično. Potrebno je istražiti i objasniti različite metode za ekstrakciju palete boja iz slika kao što su grupiranje metodom K-srednjih vrijednosti, uporaba histograma boja, analiza komponenata boje u različitim prostorima boja, frekvencijska analiza komponenata signala boje te uporaba dubokog učenja. Odabrati jednu od metoda za ekstrakciju palete boja iz slike te napraviti njezinu programsku implementaciju. Ekstrahirana paleta boja treba sadržavati dominantne boje prisutne u slici. Omogućiti dodatne funkcionalnosti po želji kao što su, na primjer, pohranjivanje rezultata u određenom formatu, mogućnost postavljanja broja boja u paleti te prijenos palete boja iz jedne slike u drugu. Provesti evaluaciju načinjenog programskog rješenja te navesti prednosti, nedostatke i moguće smjernice za budući razvoj.

Rok za predaju rada: 14. lipnja 2024.

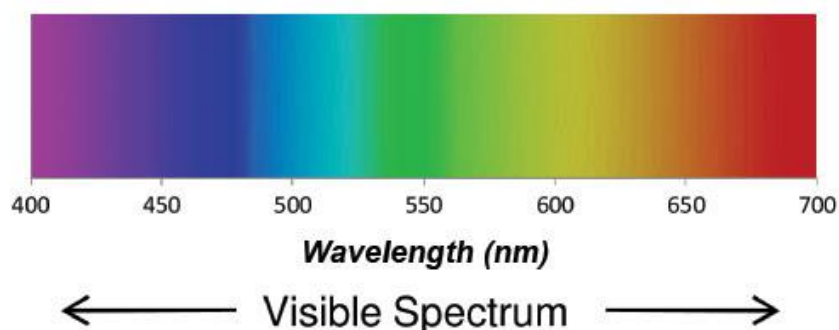
*Zahvaljujem mentorici prof. dr. sc. Sonji Grgić na suradnji u izradi završnog rada
te svima koji su me podržavali tijekom studiranja.*

SADRŽAJ

1. Uvod	3
2. Metode za ekstrakciju palete boja iz slika	5
2.1. Grupiranje metodom K–srednjih vrijednosti	5
2.2. Uporaba histograma boja.....	7
2.3. Analiza komponenta boje u RGB prostoru boja	9
2.4. Frekvencijska analiza komponenta signala boje	11
2.5. Uporaba dubokog učenja.....	12
3. Opis korištenog skupa podataka	14
4. Programska implementacija	16
4.1. Osnovna funkcionalnost.....	16
4.2. Dodatne funkcionalnosti.....	17
4.2.1. Pohranjivanje rezultata u PNG formatu	18
4.2.2. Pohranjivanje rezultata u tekstualnom formatu	19
4.2.3. Mogućnost postavljanja broja boja u paleti.....	20
4.2.4. Prijenos boja iz jedne slike u drugu	26
4.3. Evaluacija	29
Zaključak	31
Literatura	32
Sažetak	36
Summary	37

1. Uvod

Boja je aspekt bilo kojeg objekta koji se može opisati u smislu nijanse, svjetline i zasićenosti. U fizici, boja je posebno povezana s elektromagnetskim zračenjem raspona valnih duljina koje su vidljive ljudskom oku (od oko 400 do 700 nm). Zračenje takvih valnih duljina čini dio elektromagnetskog spektra poznatog kao vidljivi spektar, odnosno svjetlost. Tako primjerice svjetlost valnih duljina oko 400 nm čini ljubičastu svjetlost, a svjetlost valnih duljina oko 700 nm čini crvenu svjetlost. Na Slici 1.1 prikazan je cijeli vidljivi spektar.



Slika 1.1 Prikaz vidljivog spektra [2]

Boje utječu na način na koji doživljavamo svijet oko sebe – imaju važnu ulogu u prijenosu informacija, utječu na naše raspoloženje, potiču naše misli i osjećaje te mogu utjecati na naše odluke. Paleta boja skup je koji se sastoji od određenog broja boja. Može se koristiti u dizajnu i umjetnosti za stvaranje kohezivne i vizualno privlačne estetike. Na primjer, dizajner može odabrati određenu paletu boja za web stranicu kako bi stvorio dosljedan i profesionalan izgled. Slikar može odabrati paletu boja za umjetničko djelo kako bi prenio određeno raspoloženje ili emociju (često slike s hladnijim tonovima boja doživljavamo kao smirenije i tužnije, a slike s toplijim tonovima kao sretnije i razigranije).

U ovom radu opisan će se neke od metoda koje se mogu koristiti za ekstrakciju palete boja iz slika te će se prikazati rješenje izrade programske implementacije koja

korištenjem jedne od opisanih metoda nalazi paletu boja iz slika uz neke dodatne funkcionalnosti.

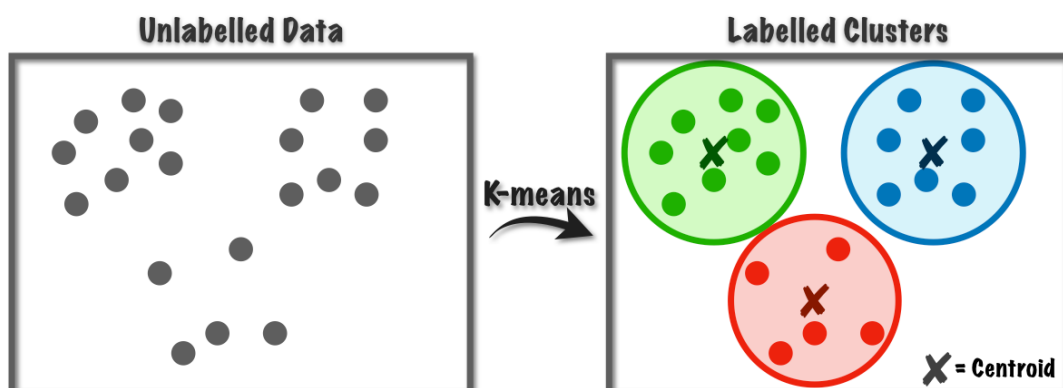
2. Metode za ekstrakciju palete boja iz slika

U nastavku će biti objašnjene neke metode koje se mogu koristiti za ekstrakciju palete boja iz slika. Konkretno, to su sljedeće metode: grupiranje metodom K-srednjih vrijednosti, uporaba histograma boja, analiza komponenta boje u različitim prostorima boja, frekvencijska analiza komponenta signala boje te uporaba dubokog učenja.

2.1. Grupiranje metodom K-srednjih vrijednosti

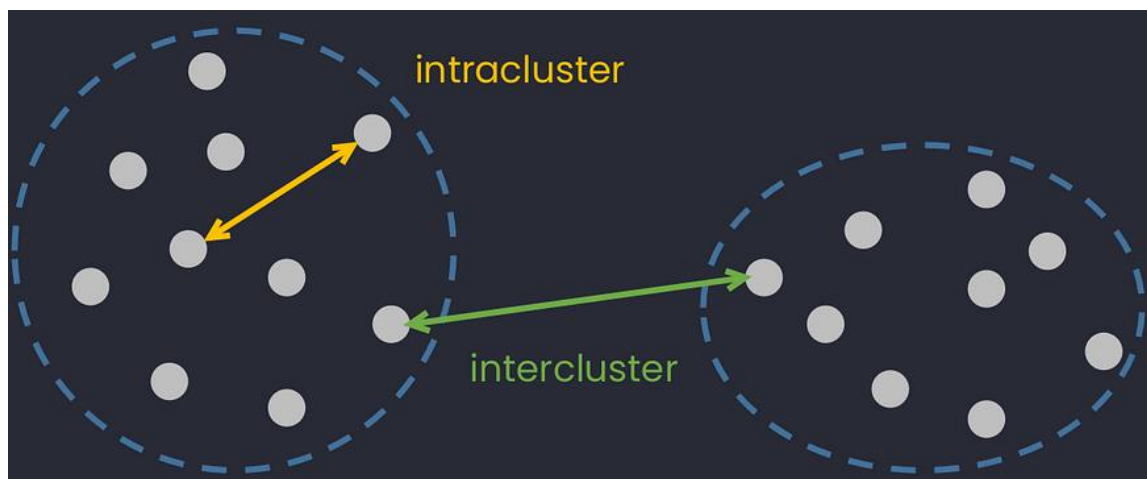
Grupiranje (engl. *clustering*) je postupak kojim se pikseli razdjeljuju u grupe na temelju sličnosti (engl. *clusters*). Cilj grupiranja nalaženje je intrinzičnih grupa u skupu neoznačenih podataka. Razlikujemo čvrsto (engl. *hard clustering*) i meko (engl. *soft clustering*) grupiranje. Pri čvrstom grupiranju jedan primjer može pripadati jednoj i samo jednoj (pod)grupi.

Grupiranje metodom K-srednjih vrijednosti najjednostavniji je i najpoznatiji algoritam čvrstog grupiranja. K predstavlja broj (čvrstih) grupa (engl. *cluster*) u koje se pikseli grupiraju na temelju međusobne sličnosti. Svaka grupa ima srednju vrijednost (centroid) kojom je predstavljena. Primjer grupa s istaknutim centroidima u slučaju kada K iznosi tri prikazan je na Slici 2.1.



Slika 2.1 Prikaz grupa pri grupiranju metodom K-srednjih vrijednosti [3]

Algoritam radi tako da se na početku nasumično izabere K centroida. U drugom koraku se svaka točka podataka dodijeli najbližem centroidu, čime će se formirati K grupa. U trećem koraku, nakon što su sve točke dodijeljene grupama, centroidi se ažuriraju. Novi centri računaju se kao srednja vrijednost svih točaka unutar njihove grupe. Drugi i treći korak ponavljaju se sve dok se centri grupa ne mijenjaju značajno ili sve dok nije dosegnut maksimalan broj iteracija ukoliko je zadan. Poželjno je algoritmom dobiti podatke grupirane tako da je razlika točaka unutar grupe što manja, a razlika između točaka koje pripadaju različitim grupama što veća – prikazano na Slici 2.2.



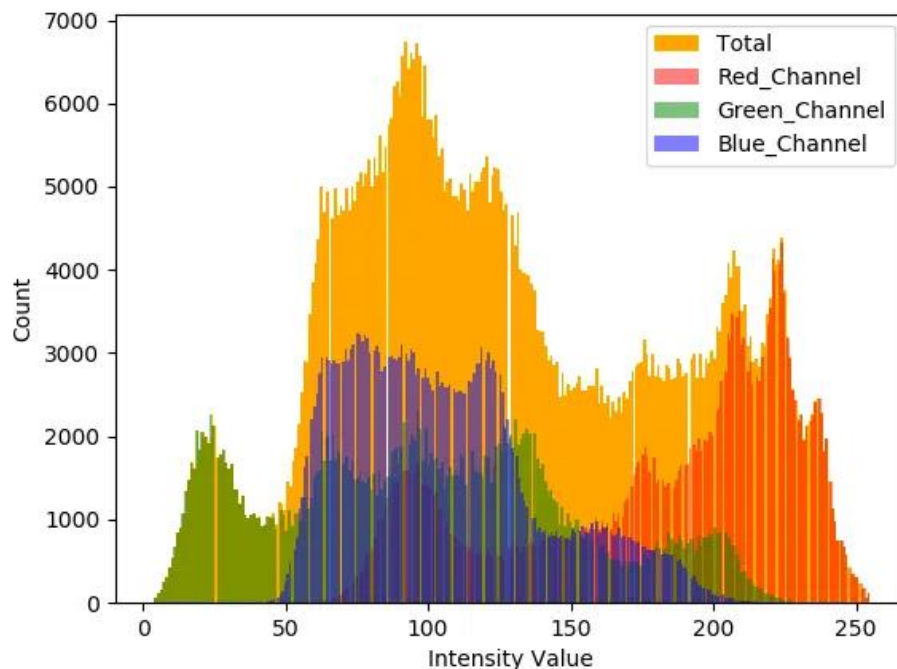
Slika 2.2 Prikaz razlike točaka unutar jedne grupe i točaka koje su u različitim grupama u algoritmu K-srednjih vrijednosti [23]

Ovaj algoritam pripada nenadziranoj vrsti strojnog učenja budući da se tijekom učenja koristi grupom neoznačenih primjera, bez informacije kojoj klasi pripada svaki pojedini primjer. Algoritam ima složenost $O(KnT)$, gdje je K broj grupa, n broj podataka (u slučaju analize slika on označava broj piksela), a T broj iteracija algoritma. Grupiranje metodom K-srednjih vrijednosti ima primjenu u raznim vrstama segmentacije (npr. segmentaciji slika) te u kompresiji slika.

2.2. Uporaba histograma boja

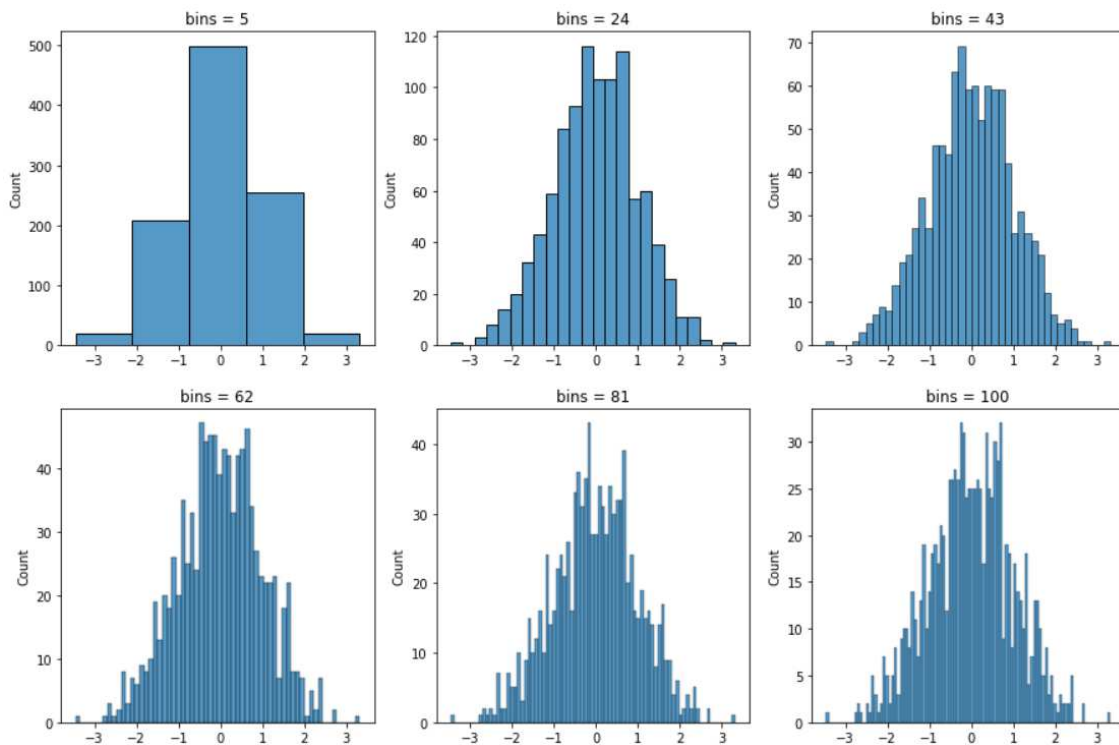
Histogram je grafički prikaz kojim se mogu prikazati distribucije frekvencija.

Histogram boje koristi se za mjeru distribucije različitih boja u slikama prilikom njihove obrade. Podaci sadržani u njemu dobivaju se brojanjem pojavljivanja svake moguće boje određenog prostora boja unutar slike. X-os histograma boja predstavlja različite vrijednosti intenziteta piksela, a y-os njihove učestalosti pojavljivanja. Primjer histograma boja u RGB prostoru boja prikazan je na Slici 2.3.



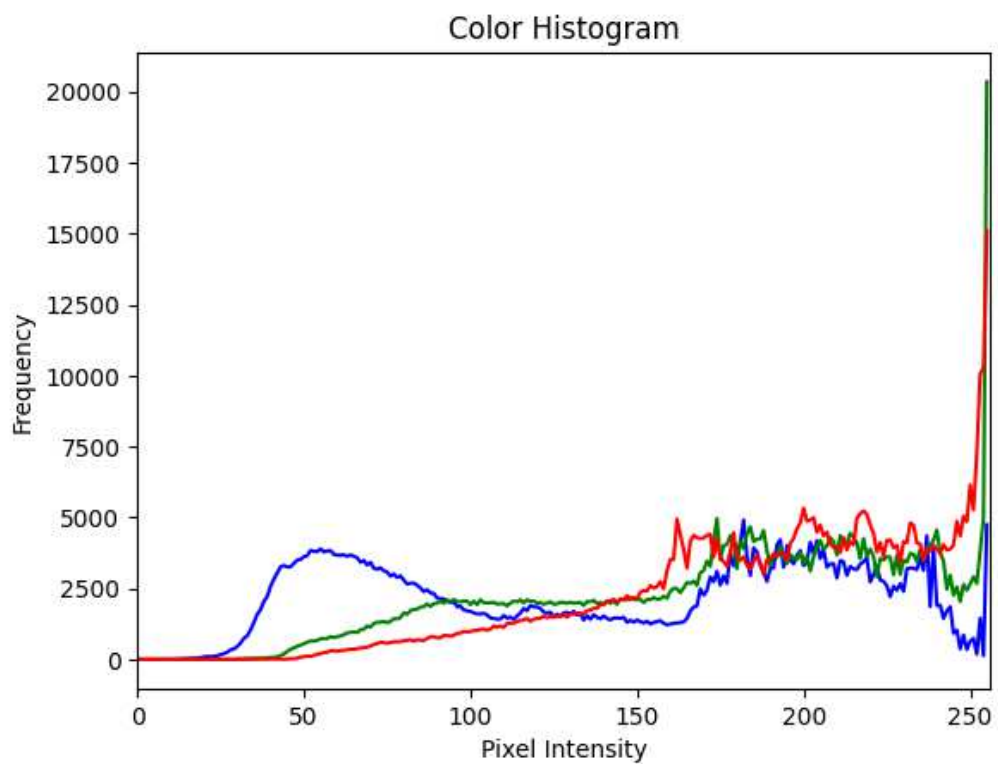
Slika 2.3 Primjer histograma boja u RGB prostoru boja [10]

Histogram boje moguće je prikazati kao niz spremnika (engl. *bin*) tako da svaki spremnik sadrži broj piksela koji su u određenom rasponu vrijednosti. Prikaz histograma boja s različitim veličinama spremnika prikazan je na Slici 2.4.



Slika 2.4 Prikaz histograma boja s različitim veličinama spremnika [9]

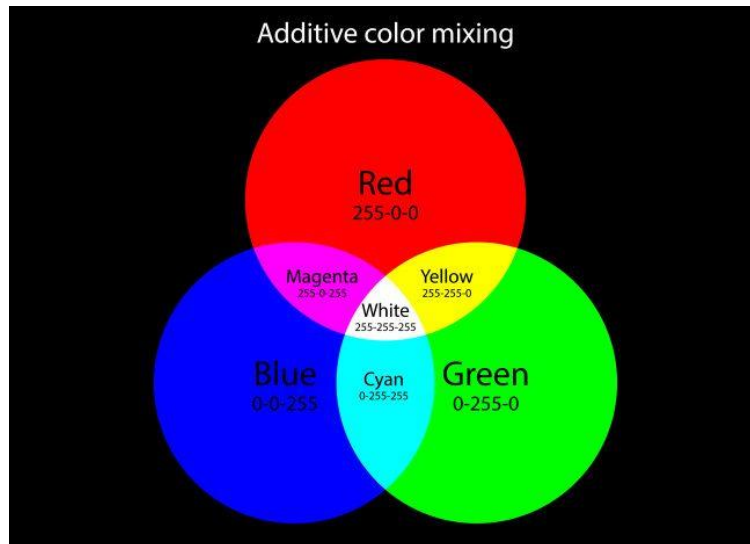
Za detaljniju analizu, često se kreiraju zasebni histogrami za svaku komponentu prostora boja. Primjerice, za RGB prostor mogu se kreirati zasebni histogrami boja za svaku od tri primarnih boja u njemu tako da je za svaku prikazan njezin intenzitet vrijednošću između 0 i 255. Prikaz takvih histograma u jednom grafu demonstriran je Slikom 2.5.



Slika 2.5 Prikaz holograma boja za primarne boje RGB prostora [9]

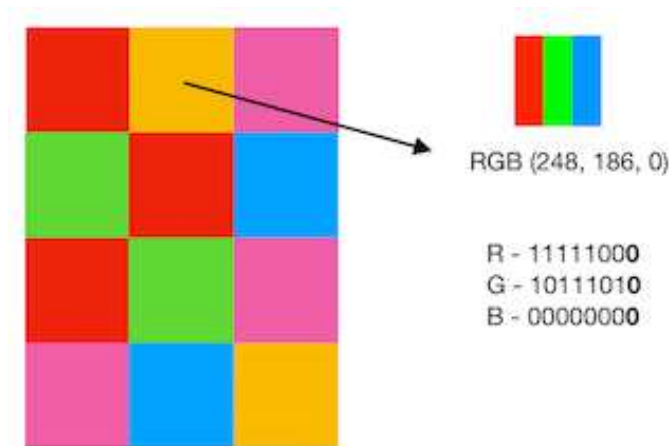
2.3. Analiza komponenata boje u RGB prostoru boja

RGB prostor boja sustav je kojim se prikazuju boje korištene na digitalnim ekranima. Aditivni je model koji se sastoji od tri primarne boje – crvene (engl. *Red*), zelene (engl. *Green*) i plave (engl. *Blue*) čijim se kombiniranjem u određenim količinama dobivaju i sve ostale boje u vidljivom spektru boja. Slikovni prikaz RGB prostora boja prikazan je na Slici 2.6.



Slika 2.6 Prikaz RGB prostora boja [12]

RGB prostor boja postoji od sredine 1800.-ih godina, a danas se intenzivno koristi u modernoj tehnologiji. U modernim sustavima svaki piksel koristi 8 bitova za svaku od tri primarne boje, tj. vrijednosti 0-255. Time je omogućen prikaz 16777216 različitih diskretnih kombinacija R, G i B vrijednosti. Primjer takve kombinacije za jedan piksel prikazan je na Slici 2.7.



Slika 2.7 Prikaz vrijednosti piksela u RGB prostoru boja [11]

2.4. Frekvencijska analiza komponenata signala boje

Frekvencijska analiza komponenata signala boje tehnika je kojom se može promatrati kako se boje mijenjaju unutar slike na temelju njihovih različitih učestalosti. Frekvencijska analiza komponenata signala boje važna je u razumijevanju vizualne istaknutosti unutar slike. Vizualna istaknutost perceptivna je kvaliteta koju imaju dijelovi slike koji se ističu u odnosu na ostali dio slike i time pridobivaju našu pažnju. Metode korištene za procjenu vizualne istaknutosti mogu se klasificirati kao biološki temeljene, računске ili kao kombinacija to dvoje. Neke metode korištene za njezino određivanje koje će se promatrati su Itti et al., Ma i Zhang, Harel et al., Hou i Zhang i Achanta et al. [21]. Frekvencijska analiza komponenata signala boje važna je za razumijevanje različitih metoda detekcije vizualne istaknutosti. Ove metode analiziraju prostorne frekvencije (brzinu promene intenziteta piksela) prisutne u slici kako bi identificirale regije koje privlače pažnju.

Metode za prepoznavanje vizualne istaknutosti često imaju različite pristupe za analizu prostornog frekvencijskog sadržaja slike. Primjerice, metoda Itti et al. koristi piramidu Gaussovih razlika (engl. *Difference of Gaussian pyramid*) za procjenu kontrasta centra i okoline čime omogućuje analizu niskih prostornog frekvencija koje odgovaraju širem kontekstu slike, dok metoda Ma i Zhang koristi prosječne vrijednosti blokova piksela za procjenu vizualne istaknutosti, zadržavajući frekvencijski raspon do $\pi/10$ originalnog signala.

Na Slici 2.8 prikazani su jednodimenzionalni frekvencijski rasponi metoda za prepoznavanje vizualne istaknutosti. Na njoj je metoda Itti et al. označena s IT, metoda Ma i Zhang s MZ, metoda Harel et al. s GB, metoda Hou i Zhang sa SR te metoda Achanta et. al s AC.

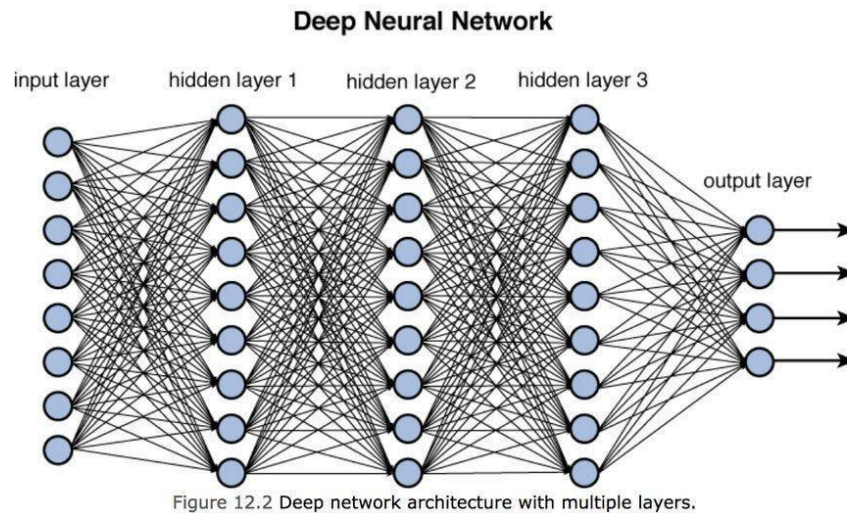
Tablica 2.1 Frekvencijski rasponi metoda za prepoznavanje vizualne istaknutosti [21]

Method	Freq. range
IT	$[\pi/256, \pi/16]$
MZ	$[0, \pi/10]$
GB	$[\pi/128, \pi/8]$
SR	$[0, \pi/5]$
AC	$(0, \pi]$

2.5. Uporaba dubokog učenja

Duboko učenje grana je strojnog učenja (grane umjetne inteligencije i računalne znanosti) koja se temelji na složenim podatkovnim reprezentacijama do kojih se dolazi nizom naučenih nelinearnih transformacija. Dubokim učenjem oponašaju se složene biološke neuronske mreže ljudskog mozga, omogućujući računalima da otkrivaju obrasce i donose odluke iz velike količine nestrukturiranih podataka.

Duboke neuronske mreže koje se koriste u metodi dubokog učenja sastoje se od više slojeva međusobno povezanih čvorova, pri čemu svaki sloj nadograđuje prethodni kako bi se njime precizirale i optimizirale predikcije i klasifikacije. Slojevi između ulaznog i izlaznog sloja nazivaju se skriveni slojevi (engl. *hidden layers*). Na Slici 2.8 slikovno je prikazana arhitektura duboke neuronske mreže s tri skrivena sloja.



Slika 2.8 Prikaz arhitekture duboke neuronske mreže s tri skrivena sloja [24]

Duboko učenje izvodi nelinearne transformacije svojih ulaznih podataka te koristi ono što je naučeno kako bi se stvorio statistički model kao izlaz. Iteracije se nastavljaju sve dok izlaz ne postigne prihvatljivu razinu točnosti. Veliki broj slojeva obrade kroz koje podaci prolaze u ovom procesu inspirirali su naziv metode duboko učenja.

Metode dubokog učenja imaju primjenu u različitim područjima umjetne inteligencije poput računalnog vida, obrade prirodnog jezika, razumijevanja govora i zvučnih signala, kao i u bioinformatici.

3. Opis korištenog skupa podataka

U ovom radu korištene su slike iz skupa podataka COCO (engl. *Common Objects in Context*). Ovaj javno dostupan skup podataka često se koristi u istraživanju računalnog vida. Skup se sastoji od više od 330000 slika. Svaka slika ima svoj jedinstveni identifikacijski broj (id), od kojih većina (njih više od 200000) ima oznake u JSON formatu koje se mogu koristiti u zadacima poput segmentacije slike, prepoznavanja objekata iz slike i stvaranja tekstnog opisa slike. U skupu podataka COCO nalaze se slike na kojima su razni objekti (primjerice ljudi, predmeti i životinje) te slike sadrže širok raspon boja.

Slike iz skupa podataka COCO preuzete su s interneta i spremljene lokalno na disk računala. Putanje do tih slika korištene su u programskoj implementaciji te se te slike koriste za učitavanje i obradu. Njima je na taj način također omogućeno testiranje rada programske implementacije. U nastavku će se u ovom radu demonstrirati funkcioniranje programske implementacije na paru slika na kojima je prikazan različit sadržaj - Slike plaža (Slika 3.1), čiji je identifikacijski broj u skupu podataka COCO 353446 i Slike zebra (Slika 3.2), čiji je identifikacijski broj 372022.



Slika 3.1 Slika plaža



Slika 3.2 Slika zebra

Također će se rad programske implementacije demonstrirati na paru paru slika na

kojima je prikazan sličan sadržaj - slike Buket 1 (Slika 3.3), čiji je identifikacijski broj u skupu podataka COCO 298370 i slike Buket 2 (Slika 3.4), čiji je identifikacijski broj 506748.



Slika 3.3 Slika buket 1



Slika 3.4 Slika buket 2

4. Programska implementacija

Programska implementacija pisana je u programsku jezik Python. Pritom se koriste razne Pythonove biblioteke koje uključuju biblioteku za numeričke proračine NumPy, biblioteku za strojno učenje Scikit-learn i biblioteku za rad s obradom slika PIL.

4.1. Osnovna funkcionalnost

Osnovna je funkcionalnost programske implementacije ekstrahirati paletu boja iz slika. To je postignuto grupiranjem metodom K-srednjih vrijednosti. Grupiranje metodom K-srednjih vrijednosti omogućuje prepoznavanje dominantnih boja u slici tako da se pikseli grupiraju u K grupa, pri čemu je K broj željenih boja u paleti. Dio koda u kojem se nalazi funkcija kojom se implementira ovaj algoritam prikazan je na Slici 4.1.

```
def k_means_extraction(arr, height, width, palette_size):
    arr = np.reshape(arr, (width * height, -1))
    model = KMeans(n_clusters=palette_size, n_init='auto', init='k-means++', random_state=42)
    labels = model.fit_predict(arr)
    palette = np.array(model.cluster_centers_, dtype=int)
    color_count = np.bincount(labels)
    color_frequency = color_count / float(np.sum(color_count))
    colors = [(color, freq) for color, freq in zip(palette, color_frequency)]
    return colors
```

Slika 4.1 Funkcija kojom je implementiran algoritam nalaženja palete boja

Za broj K može se uzeti optimalan broj grupa dobiven metodom lakta (engl. *elbow method*). Ona se koristi za nalaženjem stanja u kojem dodavanje dodatnih grupa ne bi značajno smanjilo varijance unutar grupa. Dio koda u kojem se nalazi optimalan broj grupa prikazan je na Slici 4.2.

```

def determine_optimal_palette_size(image, max_clusters=10):
    img = Image.open(image).convert("RGB")
    img = img.resize((256, 256))
    arr = np.asarray(img)
    arr = np.reshape(arr, (img.width * img.height, -1))
    distortions = []
    for i in range(1, max_clusters + 1):
        model = KMeans(n_clusters=i, n_init='auto', init='k-means++', random_state=42)
        model.fit(arr)
        distortions.append(model.inertia_)

    elbow_point = 1
    for i in range(1, len(distortions) - 1):
        if (distortions[i - 1] - distortions[i]) / (distortions[i] - distortions[i + 1] + 1e-6) < 1:
            elbow_point = i
            break

    return elbow_point + 1

```

Slika 4.2 Prikaz funkcije za nalaženje optimalnog broja K

Broj grupa korisnik programske implementacije može zadati i po želji (detaljnije opisano u poglavlju 4.2.2. Pohranjivanje rezultata u tekstualnom formatu).

Proces se sastoji od nekoliko koraka. Na početku se slika učitava, pri čemu se ona konvertira u RGB format. Nakon toga, slika se po potrebi smanjuje na dimenzije 256x256 piksela kako bi se ubrzao proces obrade. Zatim se pikseli slike transformiraju u niz koji se prosljeđuje algoritmu metode K-srednjih vrijednosti. Algoritmom se grupiraju pikseli u željeni broj (K) grupa, gdje je svaka grupa predstavljena centroidom koji odgovara jednoj boji u paleti. Nakon toga, za svaku se grupu izračunava učestalost pojavljivanja pripadajućih piksela kako bi se dobila učestalost pojavljivanja boja u slici. Na kraju, dobivene boje sortiraju se prema učestalosti pojavljivanja, a zatim se pretvaraju u heksadekadski format da bi se mogle jednostavnije prikazivati i koristiti.

4.2. Dodatne funkcionalnosti

Osim opisane osnovne funkcionalnosti, ovom programskom implementacijom omogućene su i sljedeće dodatne funkcionalnosti.

4.2.1. Pohranjivanje rezultata u PNG formatu

Korisnicima ove programske implementacije omogućena je funkcionalnost pohranjivanja rezultata ekstrakiranja palete boja po želji spremiti kao sliku. Pohranjivanje rezultata kao slike postignuto je funkcijom kojom se stvara nova slika koja prikazuje kvadratiće boja iz dobivene palete – prikazano Slikom 4.3.

```
def create_color_palette_image(colors_hex, image_path):
    width = 100 * len(colors_hex)
    height = 100
    palette_image = Image.new("RGB", (width, height))

    draw = ImageDraw.Draw(palette_image)
    for i, (color, _) in enumerate(colors_hex):
        draw.rectangle([i * 100, 0, (i + 1) * 100, 100], fill=color)

    output_image_path = os.path.splitext(image_path)[0] + "_palette.png"
    palette_image.save(output_image_path)
    palette_image.show()
    print(f"Paleta boja je spremljena u: {output_image_path}")
```

Slika 4. 3 Prikaz funkcije za pohranu rezultata u PNG formatu

Svaki je kvadratić vizualna reprezentacija jedne boje iz ekstrahirane palete, a slika se sprema u PNG formatu. U spomenutoj funkciji, širina se nove slike prilagođava broju boja u paleti, a visina ostaje jednaka neovisno o broju boja. Potom se slika sprema na disk i prikazuje korisniku. Stvorena slika kojom je vizualno prikazana izvučena paleta boja za Sliku plaža prikazana je na Slici 4.4, a za Sliku buket 1 na Slici 4.5.



Slika 4.4 Paleta s dobivenim optimalnim brojem boja za Sliku plaža (PNG)



Slika 4.5 Paleta s dobivenim optimalnim brojem boja za Sliku buket 1 (PNG)

4.2.2. Pohranjivanje rezultata u tekstualnom formatu

Rezultate korisnik po želji može pohraniti i u obliku tekstualne datoteke. Pohranjivanje rezultata u tekstualnom formatu postignuto je funkcijom kojom se stvara tekstualna datoteka koja sadrži heksadekadske vrijednosti boja zajedno s njihovim učestalošću pojavljivanja u slici. Ta funkcija prikazana je na Slici 4.6.

```
def save_palette_to_text(colors_hex, image_path):
    output_text_path = os.path.splitext(image_path)[0] + "_palette.txt"
    with open(output_text_path, 'w') as f:
        for color, freq in colors_hex:
            f.write(f"{color}: {freq:.2%}\n")
    print(f"Paleta boja je spremljena u: {output_text_path}")
```

Slika 4.6 Prikaz funkcije za pohranu rezultata u tekstualnom formatu

Boje su predstavljene u formatu #RRGGBB (crvena, zelena i plava komponenta boje, predstavljene svaka s dva heksadekadska znaka – od 0 do 255), a učestalost pojavljivanja svake boje prikazana je kao postotak zaokružen na dvije decimale. Tekstualna datoteka sprema se na disk i omogućuje korisniku jednostavan pristup

informacijama o paleti boja. Rezultat ekstrahiranja palete boja u tekstualnom formatu za Sliku Plaža prikazan je na Slici 4.7, a za Sliku buket 1 na Slici 4.8.

```
#afbbbf: 34.51%  
#4799ab: 24.82%  
#181e2a: 18.27%  
#b02a0b: 15.11%  
#026dd8: 7.29%
```

Slika 4. 7 Paleta s dobivenim optimalnim brojem boja za Sliku plaža (txt)

```
#fefdfe: 66.62%  
#26131a: 15.69%  
#8f0915: 6.82%  
#b96c27: 4.95%  
#6a4e2c: 4.07%  
#b89385: 1.85%
```

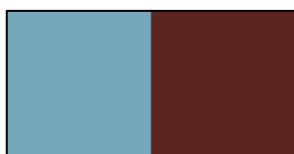
Slika 4. 8 Paleta s dobivenim optimalnim brojem boja za sliku Slika buket 1 (txt)

4.2.3. Mogućnost postavljanja broja boja u paleti

U ovoj programskoj implementaciji omogućeno je da korisnici mogu sami specificirati broj boja koje žele izvući iz slike. Koristi se algoritam metode K-srednjih vrijednosti za grupiranje piksela slike u željeni zadani broj grupa, tj. različitih boja koje korisnici mogu odrediti. Algoritam pronalazi centralne točke za svaku grupu koje predstavljaju dominantne boje u slici.

Na Slici 4.9, Slici 4.10 i Slici 4.11 prikazani su slikovni prikazi palete boja za Sliku plaža u ovisnosti o odabranom broju različitih grupa, tj. boja (na Slici 4.9 paleta s

dvije boje, na Slici 4.10 s 4 boje, a na Slici 4.11 s 8 boja).



Slika 4.9 Slikovni prikaz palete s dvije boje za Sliku plaža



Slika 4.10 Slikovni prikaz palete s 4 boje za Sliku plaža



Slika 4.11 Slikovni prikaz palete s 8 boja za Sliku plaža

Na Slici 4.12 prikazana je Slika plaža s dvije dominantne ekstrahirane boje, na Slici 4.13 s 4 dominantne ekstrahirane boje, a na Slici 4.14 s 8 dominantnih ekstrahiranih boja.



Slika 4.12 Slika plaža s dvije dominantne ekstrahirane boje



Slika 4.13 Slika plaža s 4 dominantne ekstrahirane boje



Slika 4.14 Slika plaža s 8 dominantnih ekstrahiranih boja

Na Slici 4.15, Slici 4.16 i Slici 4.17 prikazani su slikovni prikazi palete boja za Sliku buket 1 u ovisnosti o odabranom broju različitih grupa, tj. boja (na Slici 4.15 paleta s dvije boje, na Slici 4.16 s 4 boje, a na Slici 4.17 s 8 boja).



Slika 4.15 Slikovni prikaz palete s dvije boje za Sliku buket 1



Slika 4.16 Slikovni prikaz palete s 4 boje za Sliku buket 1



Slika 4.17 Slikovni prikaz palete s 8 boja za Sliku buket 1

Na Slici 4.18 prikazana je Slika buket 1 s dvije dominantne ekstrahirane boje, na Slici 4.19 s 4 dominantne ekstrahirane boje, a na Slici 4.20 s 8 dominantnih ekstrahiranih boja.



Slika 4.18 Slika buket 1 s dvije dominantne ekstrahirane boje



Slika 4.19 Slika buket 1 s 4 dominantne ekstrahirane boje



Slika 4.20 Slika buket 1 s 8 dominantnih ekstrahiranih boja

4.2.4. Prijenos boja iz jedne slike u drugu

Nakon uspješne ekstrakcije boja iz jedne (izvorne) slike, korisniku je omogućena funkcionalnost da te izvučene boje prenese na drugu (ciljanu) sliku. Program učitava ciljanu sliku i na njoj koristi algoritam K-srednjih vrijednosti kako bi se na njoj pronašle dominantne boje, ali ovoga puta s brojem grupa jednakom broju boja izvučenih iz izvorne slike. Zatim se pikseli ciljane slike zamjenjuju bojama koje su im najbližije iz izvučene palete (sličnost se mjeri u RGB prostoru boja). Ovaj problem sparivanja riješen je mađarskom metodom – algoritmom kojim se spajaju elementi iz dva skupa prema udaljenosti [25].

Nova slika dobivena prijenosom palete boja prikazane na Slici 4.9 na Sliku zebra prikazana je na Slici 4.21. Slika dobivena prijenosom palete boja prikazane na Slici 4.10 na Sliku zebra prikazana je na Slici 4.22, a na Slici 4.23 prikaz je slike dobivene prijenosom palete boja prikazane na Slici 4.11 na Sliku zebra.



Slika 4.21 Slika zebra obojana paletom s dvije boje iz Slike plaža



Slika 4.22 Slika zebra obojana paletom s 4 boje iz Slike plaža



Slika 4.23 Slika zebra obojana paletom s 8 boja iz Slike plaža

Nova slika dobivena prijenosom palete boja prikazane na Slici 4.15 na Sliku buket 2 prikazana je na Slici 4.24. Slika dobivena prijenosom palete boja prikazane na Slici 4.16 na Sliku buket 2 prikazana je na Slici 4.25, a na Slici 4.26 prikaz je slike dobivene prijenosom palete boja prikazane na Slici 4.17 na Sliku buket 2.



Slika 4.24 Slika buket 2 obojana paletom s dvije boje iz Slike buket 1



Slika 4.25 Slika buket 2 obojana paletom s 4 boje iz Slike buket 1



Slika 4.26 Slika buket 2 obojana paletom s 8 boja iz Slike buket 1

4.3. Evaluacija

Programska implementacija evaluirat će se prema procjeni točnosti dobivene paleta boja iz izvorne slike korištenjem metode K-srednjih vrijednosti te prema potrebnom vremenu izvršavanja implementacije.

Vizualnom usporedbom na najjednostavniji način može se utvrditi da palete boja za slike nađene grupiranjem K-srednjih vrijednosti dobro reprezentiraju slike iz kojih su izvučene. Uglavnom vrijedi da što je veći postavljeni broj boja u paleti, paleta će biti vjerniji prikaz dominantnih boja u slici. Programska implementacija vrlo je vremenski učinkovita, tj. brzo se izvršava čak i u slučajima kad je željeni broj boja u paleti veći. Na primjer, kad korisnik traži paletu sa 100 najčešćih boja u paleti, tražene boje pronađu se gotovo instantno.

Neki nedostaci programske implementacije proizlaze iz načina funkcioniranja grupiranja metodom K-srednjih vrijednosti. Budući da algoritam inicijalno nasumično određuje centroide, krajnji rezultat može ovisiti o početnom izboru centroida.

Nadalje, metodom lakta, nalaženje stanja u kojem dodavanje dodatnih grupa ne bi značajno smanjilo varijance unutar grupa ponekad može biti nedovoljno precizno i ovisiti o skupu podatka.

Zaključak

U ovom radu istražene su i implementirane metode za ekstrakciju palete boja iz slika. Detaljno su opisane različite metode za ekstrakciju palete boja, uključujući grupiranje metodom K-srednjih vrijednosti, uporabu histograma boja, analizu komponenata boje u različitim prostorima boja, frekvencijsku analizu komponenata signala boje te uporabu dubokog učenja.

Na temelju provedenog istraživanja, izabrana je metoda grupiranjem K-srednjih vrijednosti za implementaciju zbog njezine jednostavnosti i učinkovitosti u prepoznavanju dominantnih boja. Programska implementacija, razvijena u Pythonu, omogućava korisnicima ekstrahiranje palete boja iz slika, kao i niz dodatnih funkcionalnosti poput pohranjivanja rezultata u različitim formatima i prilagodbe broja boja u paleti. Implementacija je testirana vizualno i tekstualno na slikama iz skupa podataka COCO.

Provedenom evaluacijom utvrđene su prednosti korištenja algoritma K-srednjih vrijednosti u ekstrakciji palete boja, poput brzine i točnosti ekstrakcije, ali su prepoznati i određeni nedostaci, kao što je osjetljivost na odabir početnih centroida.

Moguće su razne nadogradnje i poboljšanja implementacije. Neke od njih su: mogućnost da se iz dobivene palete boja pretpostave osjećaji koje slika izaziva kod njezinog promatrača; mogućnost da se iz više od jedne izvorne slike pronađu dominantne boje i prenesu na ciljanu sliku te funkcionalnost da se samo odabrane boje iz palete boja prenesu na ciljanu sliku. Također bi dodatna funkcionalnost mogla biti da se dominantne boje ciljane slike zamijene dominantnim bojama izvorne slike koje su im najrazličitije (sada se mijenjaju najsličnijim bojama). Također bi se moglo ostvariti spremanje palete boja i u drugim formatima (npr. JSON ili CSV) te bi se moglo razviti grafičko korisničko sučelje (GUI) da bi korisnicima bilo olakšano služenje programskom implementacijom.

Literatura

[1] A. M. Helmetine, The Visible Spectrum: Wavelengths and Colors (7. lipnja 2024.) Poveznica:

<https://www.thoughtco.com/understand-the-visible-spectrum-608329#:~:text=The%20human%20eye%20sees%20color,because%20humans%20can%20see%20it>; pristupljeno 7. lipnja 2024.

[2] What is the visible spectrum? (25. lipnja 2015.) Poveznica:

<https://socratic.org/questions/what-is-the-visible-spectrum>; pristupljeno 7. lipnja 2024.

[3] A. Jeffares, K-means, A Complete Introduction (19. studenoga 2019.)

Poveznica: <https://towardsdatascience.com/k-means-a-complete-introduction-1702af9cd8c>; pristupljeno 7. lipnja 2024.

[4] K. Cherry, Color Psychology: Does It Affect How You Feel? (20. veljače 2024.)

Poveznica:

<https://www.verywellmind.com/color-psychology-2795824#:~:text=Color%20can%20play%20an%20important,way%20they%20ador%20their%20environments>; pristupljeno 7. lipnja 2024.

[5] K. Nassau, colour (27. svibnja 2024.) Poveznica:

<https://www.britannica.com/science/color>; pristupljeno 7. lipnja 2024.

[6] J. Šnajder, Grupiranje, 19. Grupiranje: Strojno učenje 1, UNIZG FER, ak. god. 2021./2022., J. Šnajder, predavanja, v2.1-

Poveznica: <https://www.fer.unizg.hr/download/repository/SU1-2021-P19-Grupiranje.pdf>; pristupljeno 7. lipnja 2024.

[7] N. S. Chauhan, Introduction to Image Segmentation with K-Means clustering,

(9. kolovoz 2019.) Poveznica: <https://www.kdnuggets.com/2019/08/introduction-image-segmentation-k-means-clustering.html>; pristupljeno 7. lipnja 2024.

[8] StatQuest, K-Means Clustering, (23. svibanj 2018.)

Poveznica: https://www.youtube.com/watch?v=4b5d3muPQmA&ab_channel=Stat

[QuestwithJoshStarmer](#); pristupljeno 7. lipnja 2024.

[9] H. Sajid, Demystifying Color Histograms: A Guide to Image Processing and Analysis (10. travanj 2024.) Poveznica: <https://zilliz.com/learn/demystifying-color-histograms>; pristupljeno 10. lipnja 2024.

[10] M. Sharma, Histograms in Image Processing with skimage-Python (1. studenog 2019.) Poveznica: <https://towardsdatascience.com/histograms-in-image-processing-with-skimage-python-be5938962935>; pristupljeno 10. lipnja 2024.

[11] S. Gangurde, LSB Steganography Using Pixel Locator Sequence with AES (prosinac 2020.) Poveznica: https://www.researchgate.net/publication/346669123_LSB_Steganography_Using_Pixel_Locator_Sequence_with_AES; pristupljeno 10. lipnja 2024.

[12] V. Vicente, What Does "RGB" Mean, and Why Is It All Over Tech? (11. srpnja 2021.) Poveznica: <https://www.howtogeek.com/730577/what-does-rgb-mean-and-why-is-it-all-over-tech/>; pristupljeno 10. lipnja 2024.

[13] B. Saji, Elbow method for Finding the Optimal Number of K Clusters in K-Means (5. svibanj 2024.) Poveznica: <https://www.analyticsvidhya.com/blog/2021/01/in-depth-intuition-of-k-means-clustering-algorithm-in-machine-learning/#:~:text=Elbow%20Method%20Definition&text=It%20involves%20plotting%20the%20variance,for%20analysis%20or%20model%20training.> ;pristupljeno 11. lipnja 2024.

[14] The Drawbacks of K-Means Algorithm (18. ožujak 2024.) Poveznica: <https://www.baeldung.com/cs/k-means-flaws-improvements>; pristupljeno 11. lipnja 2024.

[15] Introduction to Deep Learning (26. svibanj 2024.) Poveznica: <https://www.geeksforgeeks.org/introduction-deep-learning/>; pristupljeno 11. lipnja 2024.

- [16] A. S. Gillis, deep learning (srpanj 2023.) Poveznica: <https://www.techtarget.com/searchenterpriseai/definition/deep-learning-deep-neural-network>; pristupljeno 11. lipnja 2024.
- [17] Som, Coco dataset, What is it? and How can we use it? (14. studenog 2021.) Poveznica: <https://someshfengde.medium.com/coco-dataset-what-is-it-and-how-can-we-use-it-e34a5b0c6ecd>; pristupljeno 11. lipnja 2024.
- [18] glenn-jocher, RizwanMunawar, Laughing-q; COCO Dataset (2. lipnja 2024.) Poveznica: <https://docs.ultralytics.com/datasets/detect/coco/>; pristupljeno 11. lipnja 2024.
- [19] Computer Graphics | The RGB color model (8. srpanj 2022.) Poveznica: <https://www.geeksforgeeks.org/computer-graphics-the-rgb-color-model/>; pristupljeno 11. lipnja 2024.
- [20] RGB (red, green and blue) (siječanj 2023.) Poveznica: <https://www.techtarget.com/whatis/definition/RGB-red-green-and-blue>; pristupljeno 11. lipnja 2024.
- [21] R. Achanta, S. Hemami, F. Estrada, S. Susstrunk, Frequency-tuned salient region detection, (2009.) Poveznica: <https://ieeexplore.ieee.org/document/5206596>; pristupljeno 11. lipnja 2024.
- [22] S. Šegvić, Duboko učenje 1, (17. siječanj 2024.) Poveznica: <https://www.zemris.fer.hr/~ssegvic/du/du0intro.pdf>; pristupljeno 12. lipnja 2024.
- [23] D. Kharkar, K-Means Clustering Algorithm. (11. lipnja 2023.) Poveznica: <https://medium.com/@dishantkharkar9/k-means-clustering-algorithm-ce4fbcac8fb0>; pristupljeno 12. lipnja 2024.
- [24] R. Parmar, Training Deep Neural Networks (11. rujan 2018.) Poveznica: <https://towardsdatascience.com/training-deep-neural-networks-9fdb1964b964>; pristupljeno 12. lipnja 2024.
- [25] A. Minisini, O. Kulkov, Hungarian algorithm for solving the assignment

problem (13. prosinac 2023.) Poveznica: <https://cp-algorithms.com/graph/hungarian-algorithm.html>; pristupljeno 14. lipnja 2024.

Sažetak

EKSTRAKCIJA PALETE BOJA IZ SLIKA

Paleta boja ima važnu ulogu u umjetnosti, fotografiji i dizajnu budući da može znatno utjecati na način kako doživljavamo slike. Ovim radom napravljen je pregled i opis metoda za ekstrakciju palete boja iz slika, a u praktičnom dijelu odabrano je grupiranje metodom K-srednjih vrijednosti za implementaciju algoritma ekstrakciju palete boja. Programski alat omogućuje korisniku ekstrahiranje dominantnih boja iz slike, pohranjivanje rezultata u različitim formatima te prilagođavanje broja boja u paleti. Također je implementirana funkcionalnost prijenosa boja iz jedne slike na drugu. Evaluacija programskog rješenja provedena je korištenjem slika iz skupa podataka COCO, a rezultati su pokazali da odabrana metoda pretežno uspješno identificira i prikazuje dominantne boje u slikama.

Ključne riječi: digitalna obrada slike, ekstrakcija palete boja, grupiranje metodom K-srednjih vrijednosti

Summary

COLOR PALETTE EXTRACTION FROM IMAGES

The color palette plays an important role in art, photography, and design as it can significantly influence the way we perceive images. This paper provides an overview and description of methods for extracting a color palette from images. In the practical part, the K-means clustering method was selected for the implementation of the color palette extraction algorithm. The software tool allows users to extract dominant colors from an image, save the results in various formats, and adjust the number of colors in the palette. Additionally, a functionality for color transfer from one image to another has been implemented. The evaluation of the software solution was conducted using images from the COCO dataset, and the results showed that the selected method predominantly successfully identifies and displays the dominant colors in the images.

Keywords: digital image processing, color palette extraction, K-means clustering