

Filip Marijanović

**UTICAJ IZBORA KOLORNOG MODELA NA  
SEGMENTACIJU SLIKE PRIMJENOM  
K-MEANS I OTSUOVOG METODA**

- master rad -

Podgorica, 2024.

UNIVERZITET CRNE GORE  
ELEKTROTEHNIČKI FAKULTET

Filip Marijanović

**UTICAJ IZBORA KOLORNOG MODELA NA  
SEGMENTACIJU SLIKE PRIMJENOM  
K-MEANS I OTSUOVOG METODA**

- master rad -

Podgorica, 2024.

## **PODACI I INFORMACIJE O STUDENTU**

Ime i prezime:

**Filip Marijanović**

Datum i mjesto rođenja:

08.04.1999. godine, Berane

Naziv završenog osnovnog studijskog programa i godina završetka studija:

Studijski program Primijenjenog računarstva, Elektrotehnički fakultet, Univerzitet Crne Gore, 180 ECTS kredita, 2021. godine.

## **INFORMACIJE O MASTER RADU**

Naziv master studija:

Master studije primijenjenog računarstva

Naslov rada:

Uticaj izbora kolornog modela na segmentaciju slike primjenom *K-means* i Otsuovog metoda

Fakultet na kojem je rad odbranjen:

Elektrotehnički fakultet

## **UDK, OCJENA I ODBRANA MASTER RADA**

Datum prijave magistarskog rada:

*20.09.2023. godine*

Datum sjednice Vijeća na kojoj je prihvaćena tema:

*16.11.2023. godine*

Komisija za ocjenu/odbranu rada:

Prof. dr Žarko Zečević, ETF Podgorica, predsjednik  
Doc. dr Miloš Brajović, ETF Podgorica, mentor  
Doc. dr Andela Draganić, ETF Podgorica, član

Mentor:

Doc. dr Miloš Brajović

Datum odbrane:

*23.09.2024. godine*

Ime i prezime autora: Filip Marijanović, BApp

## ETIČKA IZJAVA

U skladu sa članom 22 Zakona o akademskom integritetu i članom 18 Pravila studiranja na master studijama, pod krivičnom i materijalnom odgovornošću, izjavljujem da je master rad pod naslovom

### "UTICAJ IZBORA KOLORNOG MODELA NA SEGMENTACIJU SLIKE PRIMJENOM K-MEANS I OTSUOVOG METODA"

moje originalno djelo.

Podnositelj izjave,

Filip Marijanović

Filip Marijanović, BApp

U Podgorici, dana 10.06.2024. godine

# Izvod teze

U radu se istražuje segmentacija slika i zašto je važno obratiti pažnju na ulogu kolornih modela i algoritama. U radu je predstavljen teorijski osnov segmentacije. Nakon toga, opisani su kolorni modeli i njihov značaj u gledanju i razlikovanju različitih djelova slika. Segmentacija koja se istražuje u radu je fokusirana na korišćenje osnovnih algoritama segmentacije, kao što su  $K$ -means, Otsuov metod i Otsuov metod sa višestrukim postavljanjem praga, koristeći Python alate kao što su *OpenCV*, *NumPi* i drugi. Empirijska analiza se sprovodi kroz tri različita skupa podataka: prvi sadrži originalne slike, drugi uključuje slike sa različitim kontrastom i osvjetljenjem, a treći prikazuje slike koje sadrže različite objekte. Evaluacija efektivnosti uspješne segmentacije se vrši uz pomoć metrika. Pouzdanost ovih pristupa za segmentaciju potvrđena je rezultatima, koji takođe ukazuju na primjenljivost u različitim okolnostima.

**Ključne riječi:** segmentacija slike, kolorni modeli, algoritmi segmentacije,  $K$ -means, Otsuov metod, Otsuov metod sa višestrukim pragovanjem, Python, metrike

# Abstract

The thesis explores image segmentation and emphasizes the importance of considering the role of color models and algorithms. The theoretical foundation of segmentation is presented in the paper. Following that, basic color models and their significance in viewing and distinguishing different parts of images are described. The segmentation studied in the research focuses on using fundamental segmentation algorithms such as *K*-means and Otsu's method, employing Python tools like OpenCV, NumPy, and others. Empirical analysis is conducted across three different datasets: the first contains original images, the second includes images with varying contrasts and lighting, and the third showcases images containing different objects. The effectiveness of successful segmentation is evaluated using metrics. The reliability of these segmentation approaches is confirmed by the results, which also indicate their applicability in various circumstances.

**Keywords:** image segmentation, color models, segmentation algorithms, *K*-means, Otsu's method, Otsu's method with multiple thresholding, Python, metrics

# Sadržaj

Izvod teze .....	1
Abstract .....	2
Slike.....	5
Tabele .....	7
Uvod.....	9
<b>1. SEGMENTACIJA SLIKE .....</b>	<b>15</b>
<b>1.1 Tehnike segmentacije slike .....</b>	<b>16</b>
<b>1.2 Metod praga .....</b>	<b>17</b>
<b>1.2.1 Otsuov metod.....</b>	<b>18</b>
<b>1.2.1.1 Otsuov metod sa višestrukim pragovima.....</b>	<b>23</b>
<b>1.3 Metod grupisanja - klasterizacija .....</b>	<b>24</b>
<b>1.3.1 K-means.....</b>	<b>25</b>
<b>1.3.1.1 Funkcionalni principi K-means algoritma.....</b>	<b>25</b>
<b>1.3.2 Segmentacija slike pomoću K-means .....</b>	<b>28</b>
<b>2. KOLORNI MODELI.....</b>	<b>30</b>
<b>2.1 RGB.....</b>	<b>30</b>
<b>2.2 HSV .....</b>	<b>32</b>
<b>2.3 CMYK.....</b>	<b>33</b>
<b>2.4 Konverzija i reprezentacija slike u kolornim modelima.....</b>	<b>34</b>
<b>2.4.1 Konverzija i reprezentacija slike u RGB modelu .....</b>	<b>34</b>
<b>2.4.2 Konverzija i reprezentacija slike u HSV modelu.....</b>	<b>35</b>
<b>2.4.3 Konverzija i reprezentacija slike u CMYK modelu .....</b>	<b>37</b>
<b>3. Metode validacije.....</b>	<b>39</b>
<b>3.1 Dice koeficijent.....</b>	<b>40</b>
<b>3.2 F-mjera .....</b>	<b>41</b>
<b>3.2 Jaccard-ov indeks .....</b>	<b>44</b>

<b>3.4 Varijacija informacija .....</b>	46
<b>4. Eksperimentalni rezultati .....</b>	48
<b>4.1 K-means .....</b>	49
<b>4.1.1 Rezultati uspješnosti segmentacije .....</b>	49
<b>4.1.2 Rezultati uspješnosti segmentacije slika sa prilagođenim kontrastom i osvjetljenjem.....</b>	53
<b>4.1.3 Rezultati uspješnosti segmentacije slika upotrebom datasetova sa različitim objektima .....</b>	57
<b>4.2 Otsuov metod .....</b>	62
<b>4.2.1 Rezultati uspješnosti segmentacije .....</b>	62
<b>4.2.2 Rezultati uspješnosti segmentacije slika sa prilagođenim kontrastom i osvjetljenjem.....</b>	64
<b>4.2.3 Rezultati uspješnosti segmentacije slika upotrebom datasetova sa različitim objektima .....</b>	66
<b>4.3 Otsuov metod sa višestrukim pragovima .....</b>	69
<b>4.3.1 Rezultati uspješnosti segmentacije .....</b>	69
<b>4.3.2 Rezultati uspješnosti segmentacije slika sa prilagođenim kontrastom i osvjetljenjem.....</b>	73
<b>4.3.3 Rezultati uspješnosti segmentacije slika upotrebom datasetova sa različitim objektima .....</b>	77
<b>Zaključak .....</b>	82
<b>Appendix (dodatak).....</b>	83
<b>Literatura .....</b>	88

# Slike

Slika 1. Lijeva kolona sadrži originalne slike, a desna slike koje predstavljaju rezultat segmentacije .....	15
Slika 2. Klasifikacija tehnika za segmentaciju slike .....	16
Slika 3. Ilustracija jednostavnog metoda segmentacije slike zasnovanog na postavljanju praga za selekciju pripadnosti piksela jednoj od dvije moguće klase .....	17
Slika 4. Primjena Otsuovog metoda u segmentaciji digitalnih slika. Prva kolona – kolor slika, druga kolona – sivoskalirana slika dobijena na osnovu originalne slike u boji, treća kolona – rezultat segmentacije primjenom Otsuovog metoda .....	22
Slika 5. Prva kolona: originalne slike, druga kolona – slike segmentirane u 3 klase, treća kolona – slike segmentirane u 5 klase.....	23
Slika 6. Ilustracija centroida na početku K-means algoritma.....	26
Slika 7. Odabir klastera po najmanjem rastojanju i centralizovanje centroida .....	27
Slika 8. Optimalan centar .....	28
Slika 9. Prva kolona: originalne slike, druga kolona – slike segmentirane u $K = 3$ nivoa, treća kolona – slike segmentirane u $K = 6$ nivoa, četvrta kolona – slike segmentirane u $K = 10$ nivoa	29
Slika 10. Ilustracija RGB kolornog modela .....	30
Slika 11. Originalna slika (lijevo) sa R, G i B komponentama (desno).....	31
Slika 12. Kombinacije boja – RGB .....	31
Slika 13. HSV kolorni model .....	32
Slika 14. HSV kolorni model i njegovi odgovarajući kanali .....	33
Slika 15. CMYK kolorni model .....	33
Slika 16. Primjer ground-truth-a .....	39
Slika 17. Ilustracija formule za koeficijent Dice .....	41
Slika 18. Ilustracija F-Mjera metrike .....	43
Slika 19. Ilustracija izračunavanja Jaccardovog indeksa .....	45
Slika 20. Ilustracija metrike varijacije informacija .....	47
Slika 21. Rezultati uspješnosti segmentacije kod K-means algoritma pri $K = 3$ (lijevo) i pri $K = 5$ (desno) na dataset-u originalnih slika.....	50
Slika 22. Segmentirane slike pomoću kolornih modela i K-means algoritma pri $K = 3$ .....	51
Slika 23. Segmentirane slike pomoću kolornih modela i K-means algoritma pri $K = 5$ .....	52
Slika 24. Rezultati uspješnosti segmentacije kod K-means algoritma pri $K = 3$ (lijevo) i pri $K = 5$ (desno) na dataset-u sa prilagođenim kontrastom i osvjetljenjem .....	54

Slika 25. Segmentirane slike pomoću kolornih modela i K-means algoritma pri K=3 na.....	55
Slika 26. Segmentirane slike pomoću kolornih modela i K-means algoritma pri K=5 na.....	56
Slika 27. Rezultati uspješnosti segmentacije kod K-means algoritma pri K = 3 (lijevo) i pri K = 5 (desno) – unutrašnja okruženja .....	58
Slika 28. Rezultati uspješnosti segmentacije kod K-means algoritma pri K = 3 (lijevo) i pri K = 5 (desno) – prirodna okruženja.....	59
Slika 29. Segmentirane slike pomoću kolornih modela i K-means algoritma pri K=3 – prirodna i unutrašnja okruženja. ....	60
Slika 30. Segmentirane slike pomoću kolornih modela i K-means algoritma pri K=5 – prirodna i unutrašnja okruženja. ....	61
Slika 31. Rezultati uspješnosti segmentacije kod Otsuovog metoda na dataset-u originalnih slika .....	63
Slika 32. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda na dataset-u originalnih slika.....	63
Slika 33. Rezultati uspješnosti segmentacije kod Otsuovog metoda na dataset-u sa prilagođenim kontrastom i osvjetljenjem .....	65
Slika 34. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda na.....	65
Slika 35. Rezultati uspješnosti segmentacije kod Otsuovog metoda – unutrašnja okruženja (lijevo) i prirodna okruženja (desno).....	67
Slika 36. Segmentirane slike pomoću kolornih modela i Otsuovog metoda – prirodna i unutrašnja okruženja. ....	68
Slika 37. Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima - 3 klase (lijevo), 5 klasa (desno) na dataset-u originalnih slika .....	70
Slika 38. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda sa višestrukim pragovima (3 klase) na dataset-u originalnih slika.....	71
Slika 39. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda sa višestrukim pragovima (5 klasa) na dataset-u originalnih slika.....	72
Slika 40. Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima - 3 klase (lijevo), 5 klasa (desno) na dataset-u čije su slike prilagođene promjenama kontrasta i osvjetljenja .....	74
Slika 41. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda sa višestrukim pragovima (3 klase) na dataset-u čije su slike prilagođene promjenama kontrasta i osvjetljenja.	75
Slika 42. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda sa višestrukim pragovima (5 klasa) na dataset-u čije su slike prilagođene promjenama kontrasta i osvjetljenja.	76

Slika 43. Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima - 3 klase (lijevo), 5 klasa (desno) – unutrašnja okruženja .....	78
Slika 44. Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima - 3 klase (lijevo), 5 klasa (desno) – prirodna okruženja .....	80
Slika 45. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda sa višestrukim pragovima (3 klase) – prirodna i unutrašnja okruženja.....	80
Slika 46. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda sa višestrukim pragovima (5 klasa) – prirodna i unutrašnja okruženja.....	81

## Tabele

Tabela 1: Rezultati uspješnosti segmentacije kod K-means algoritma pri $K = 3$ na dataset-u originalnih slika.....	49
Tabela 2: Rezultati uspješnosti segmentacije kod K-means algoritma pri $K = 5$ na dataset-u originalnih slika.....	50
Tabela 3: Rezultati uspješnosti segmentacije kod K-means algoritma pri $K = 3$ na dataset-u sa prilagođenim kontrastom i osvjetljenjem.....	53
Tabela 4: Rezultati uspješnosti segmentacije kod K-means algoritma pri $K = 5$ na dataset-u sa prilagođenim kontrastom i osvjetljenjem.....	53
Tabela 5: Rezultati uspješnosti segmentacije kod K-means algoritma pri $K = 3$ – unutrašnja okruženja .....	57
Tabela 6: Rezultati uspješnosti segmentacije kod K-means algoritma pri $K = 5$ – unutrašnja okruženja .....	58
Tabela 7: Rezultati uspješnosti segmentacije kod K-means algoritma pri $K = 3$ – prirodna okruženja .....	58
Tabela 8: Rezultati uspješnosti segmentacije kod K-means algoritma pri $K = 5$ – prirodna okruženja .....	59
Tabela 9: Rezultati uspješnosti segmentacije kod Otsuovog metoda na dataset-u originalnih slika .....	62
Tabela 10: Rezultati uspješnosti segmentacije kod Otsuovog metoda na dataset-u sa prilagođenim kontrastom i osvjetljenjem.....	64
Tabela 11: Rezultati uspješnosti segmentacije kod Otsuovog metoda – unutrašnja okruženja ...	66
Tabela 12: Rezultati uspješnosti segmentacije kod Otsuovog metoda – prirodna okruženja .....	66

Tabela 13. Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (3 klase) na dataset-u originalnih slika.....	69
Tabela 14: Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (5 klasa) na dataset-u originalnih slika.....	70
Tabela 15: Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (3 klase) na dataset-u čije su slike prilagođene promjenama kontrasta i osvjetljenja .....	73
Tabela 16: Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (5 klasa) na dataset-u čije su slike prilagođene promjenama kontrasta i osvjetljenja .....	73
Tabela 17: Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (3 klase) – unutrašnja okruženja .....	77
Tabela 18: Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (5 klasa) – unutrašnja okruženja .....	78
Tabela 19: Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (3 klase) – prirodna okruženja .....	79
Tabela 20: Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (5 klasa) – prirodna okruženja .....	79

# Uvod

U ovom radu izučava se uticaj izbora kolornog modela na segmentaciju slike primjenom *K-means* i Otsuovog metoda.

Segmentacija digitalne slike je proces koji se bavi razdvajanjem slika na manje segmente ili regije kako bi se identifikovali različiti elementi ili objekti unutar slike. Rad stavlja fokus na uticaj kolornih modela – RGB (engl. red, green, blue), HSV (engl. hue, saturation, value) i CMYK (engl. cyan, magenta, yellow, key) na segmentaciji digitalnih slika. Razmatraju se tri algoritma za segmentaciju (*K-means*, Otsuov metod i Otsuov metod sa višestrukim pragovanjem). Kolorni modeli su ključni jer omogućavaju analizu boja u slikama. Cilj je istražiti kako kolorni modeli utiču na preciznost segmentacije i na performanse odabralih algoritama za segmentaciju digitalnih slika, koji koriste informacije o boji piksela, odnosno, parametre kolornih modela.

U radu će posebna pažnja biti posvećena algoritmima *K-means* i Otsuov metod. *K-means* algoritam igra ključnu ulogu u segmentaciji digitalnih slika kada se kombinuje sa kolornim modelima. On olakšava segmentaciju tako što efikasno grupiše piksele sličnih boja u različite klastere, sve unutar odabralih kolornih modela. Ovaj proces grupisanja, koji izdvaja piksele koji dijeli sličnosti u boji, omogućava algoritmu da istakne objekte i regije unutar slike na osnovu njihovih specifičnih boja. Kao rezultat toga, *K-means* algoritam značajno poboljšava preciznost i kvalitet segmentacije digitalnih slika baziranih na bojama.

Otsuov metod je još jedan od poznatih algoritama za segmentaciju digitalnih slika koji može biti primijenjen uz kolorne modele. Ovaj metod pojednostavljuje odabir idealnog praga za binarizaciju slike, omogućavajući precizno odvajanje objekata od pozadine na osnovu intenziteta piksela. Kada se primjeni uz kolorne modele, Otsuov metod prvo predstavi sliku u odabranom prostoru boja, zatim je konvertuje u sliku sa sivim tonovima i identificira optimalni prag koji maksimalno razdvaja objekte od pozadine. Kroz primjenu ovog praga na sliku sa sivim tonovima, algoritam pruža precizne rezultate segmentacije. Otsuov metod koji radi sa više klasa se naziva Otsuov metod sa višestrukim pragovanjem.

## Predmet istraživanja

Predmet istraživanja u radu je uticaj kolornih modela na performanse segmentacije i digitalnih slika fokusirajući se na interakciju između kolornih modela i tri istaknuta algoritma – *K-means*, Otsuov metod i Otsuov metod sa višestrukim pragovanjem. Procjenom njihovih performansi pri primjeni različitih kolornih modela (RGB, HSV, CMYK) ispitivaće se prednosti i slabosti svake kombinacije, bacajući svjetlost na njihovu prikladnost za specifične zadatke segmentacije digitalne slike.

Implementacija algoritama i eksperimentalna validacija rezultata je sprovedena u programskom jeziku Python. Rezultati segmentacije i komparativna analiza su sprovedeni korišćenjem Python biblioteka za obradu slike i analizu podataka i individualnih implementacija razmatranih algoritama. Analiza će biti sprovedena nad pažljivo odabranim dataset-ovima, prikladno label-ovanim za validaciju rezultata segmentacije.

Glavne biblioteke korišćene u programskom jeziku Python su:

1. *OpenCV* - biblioteka za obradu slika
2. *Matplotlib* – biblioteka za kreiranje statičnih, animiranih i interaktivnih vizualizacija
3. *NumPy* – biblioteka za rad s nizovima i matricama

## Motivi i ciljevi istraživanja

Segmentacija digitalnih slika je jedan od najznačajnijih elemenata u domenu računarskog vida, gdje naučnici rade na razvoju tehnika koje će omogućiti bolje razumijevanje vizuelnih podataka. Ovaj proces teži da omogući računarima da „razumiju“ složene vizuelne informacije na način koji odgovara ljudskom vizuelnom zapažanju. Istraživanja u oblasti su usmjereni na segmentaciju kako bi se unaprijedila sposobnost računara, aplikacija, programa da prepoznaju, analiziraju i reaguju na različite elemente unutar slika. Međutim, izbor algoritama i kolornih modela značajno utiče na kvalitet segmentacije. U radu se detaljno analizira segmentaciju digitalne slike uz pomoć tri ključna algoritma segmentacije, *K-means*, Otsuov metod, Otsuov metod sa višestrukim pragovanjem i različitih kolornih modela.

Ključni razlozi istraživanja su:

1. Analiza kolornih modela ima važnu ulogu u istraživanju segmentacije digitalnih slika u boji i na performanse razmatranih algoritama.
2. Istraživanje prilagodljivosti na različite scenarije slike
3. Jedan od primarnih ciljeva ovog istraživanja je analiza algoritama.

Ključni motivi istraživanja su:

1. Demonstracija da je segmentacija slike pomoću kolornih modela jedna od boljih metoda za analizu vizuelnih podataka, ističući sposobnost razdvajanja i identifikacije različitih elemenata unutar slika.
2. Pružanje relevantnih informacija o performansama razmatranih algoritama u zavisnosti od izbora kolornih modela, te pokazivanje njihovih sposobnosti prilagođavanja.

## Pregled dosadašnjih istraživanja

Budući da nije postignut opšti konsenzus o tome koji je kolorni model najefikasniji za određene pristupe segmentaciji slika, neka istraživanja su bila usredsređena na identifikaciju optimalnog izbora prostora boja za specifične zadatke.

Različiti istraživači su pristupili pitanju izbora optimalnog prostora boja za segmentaciju slika na različite načine [1] [2]. Na primjer, jedna studija je fokusirana na identifikaciju najboljeg prostora boja za otkrivanje boje kože na slici [3]. Rezultati ove studije su jasno ukazali da se HSV model boja najbolje pokazao pri segmentaciji slika zasnovanoj na klasterizaciji.

S druge strane, Jurio i njegovi koautori su sproveli studiju koja analizira uticaj različitih prostora boja u segmentaciji slika zasnovanoj na klasterima [4]. Njihovo istraživanje je obuhvatilo testiranje četiri prostora boja, uključujući RGB, HSV, CMY i YUV. Njihova analiza je pokazala da su najbolji rezultati često postignuti korišćenjem CMY prostora boja, dok je i HSV pružio dobre rezultate u mnogim slučajevima.

Busin i ostali autori su predstavili metod za automatski izbor specifičnog prostora boja među klasičnim prostorima boja [5]. Ovaj odabir je urađen prema kriterijumu evaluacije zasnovanom na spektralnoj analizi boja. Njihov kriterijum procjenjuje kvalitet segmentacije u svakom prostoru i bira onaj najbolji, koji zadržava svoja specifična svojstva.

Kroz uporedne analize, Ruiz i njegovi koautori su zaključili da je za obradu u realnom vremenu u sektoru poljoprivrede za segmentaciju najbolji izbor bio HSV model boja [6]. Ovi rezultati ukazuju na to da izbor prostora boja zavisi od konkretnog zadatka, odnosno koji algoritam i koji kolorni model će se koristiti, i da nema univerzalnog rešenja.

U radu [7] je razvijen inovativan pristup segmentaciji slika u boji koristeći RGB prostor boja. Ovaj metod odvaja hromatičnost i intenzitet, što oponaša ljudsku percepciju boja i smanjuje osjetljivost na varijacije intenziteta u RGB prostoru. Korišćen je samo mali skup uzorka boja za obuku neuronske mreže za obradu nijansi boja. Istovremeno, kreiran je kanal intenziteta i primijenjen Otsuov metod za podešavanje pragova za podjelu intenziteta. Rezultati su pokazali konkurentne performanse u poređenju sa sličnim radovima.

V. Rajinikanth i M.S. Couceiro [8] predstavili su optimalnu segmentaciju slika putem Firefly algoritma (FA). U radu se koristi RGB model boja. Postizanje optimalnih pragova za svaku komponentu boje ostvaruje se maksimizacijom Otsuove funkcije izme. Predloženi postupak segmentacije demonstriran je na standardnom RGB datasetu i validiran je korišćenjem postojećih FA biblioteka. Fireflz algoritam u kombinaciju sa RGB modelom je značajno poboljšao rezultate u poređenju s drugim metodama.

Dan Burdescu i njegov koautori predstavili su segmentaciju slika u HSV prostoru boja. Korišćenje ovog modela boja poboljšava se efikasnost komponenti boje pri razdvajaju brojnih segmenata, rješavajući mnoge poteškoće u segmentaciji slika u boji. Metoda pokazuje manju vremensku složenost u poređenju s drugim metodama, a eksperimentalni rezultati na datasetovima potvrđili su uspješnost metode. [9]

Jedan od zanimljivih radova jeste detekcija akni na licu zasnovana na HSV modelu pomoću segmentacije i dubokog učenja. Akne su dermatološka bolest uzrokovana bakterijama i viškom masnoće u folikulima dlake, često rezultirajući pojavom bijelih glavica, crnih mitesera i bubuljica na koži. U ovom radu Neha Yadav i njegovi koautori su primijenili HSV kao ključni alat za segmentaciju slika lica s aknama. Ovaj model omogućava precizno izdvajanje područja zahvaćenih aknama putem detekcije nijansi boje (Hue) i zasićenosti (Saturation), što doprinosi efikasnoj obradi segmenata slike sa aknama na koži. U istraživanju koje je obuhvatilo studiju korišćenja HSV modela za segmentaciju slika lica s aknama, postignuta je impresivna tačnost od 97.54% na osnovu analize različitih datasetova.[10]

U radu [11] je predstavljen numerički efikasan algoritam za konverziju između RGB i HSV reprezentacija boja, baziran na cijelobrojnim operacijama sa cijelim brojevima umjesto realnim. Ovaj numerički efikasan algoritam nudi dvostruko ubrzanje u odnosu na klasične implementacije u programskom jeziku C++ na PC platformama, bez upotrebe tabelarnih pristupa i sa minimalnim

zahtjevima za memorijom. Rezultujuća HSV reprezentacija boja može se kodirati u samo 48 bita, čime se obezbjeđuje visok nivo tačnosti.

GrabCut se opisuje kao moćna tehnika koja nalazi primjenu u različitim aspektima segmentacije slika u boji. Ova tehnika je primijenjena u raznim domenima segmentacije, uključujući segmentaciju ljudskog tijela, segmentaciju videa, semantičku segmentaciju i segmentaciju volumena. Nedavno je razvijena potpuno paralelizovana šema za 3D segmentaciju pomoću GrabCut metode, koja je prilagođena za rad na grafičkim procesorima (GPU). [12]

## Doprinosi

Neki od doprinsa master rada su:

1. U radu je prezetovana analiza uticaja kolornih modela na različite tipove segmentacije i na različite klase slika.
2. Rezultati prezentovani u radu mogu poslužiti kao osnov za identifikaciju prednosti i nedostataka algoritama upoređivanjem rezultata segmentacije digitalnih slika na kolorne modele.
3. Rezultati dobijeni testiranjem algoritama za segmentaciju digitalnih slika uz pomoć datasetova koriste se kao osnova za odabir kolornog modela u kombinaciji s odgovarajućim algoritmom.

## Struktura rada

U uvodnom dijelu su definisani osnovni motivi i ciljevi istraživanja i uticaj kolornih modela u segmentaciji digitalnih slika. Takođe, dat je pregled evolucije i razvoja segmentacije slika kroz istoriju, gdje će se isticati ključni prethodni radovi i njihov doprinos. U sekciji je objašnjen doprinos rada, gdje će se naglasiti kako se studija uklapa u postojeće istraživačke napore u oblasti segmentacije digitalnih slika i uticaj kolornih modela.

U prvoj glavi dat je teorijski pregled osnovnih koncepata segmentacije slika. U glavi je dato i objašnjenje primjena algoritama pri segmentaciji slike.

U drugoj glavi dat je pregled kolornih modela, njihovih specifičnosti i primjera, sa posebnim osvrtom na njihov značaj za proces segmentacije. Poseban akcenat je stavljen napotreba za kolornim modelima u segmentaciji digitalnih slika, uz detaljnu specifikaciju i objašnjenje njihove uloge u razdvajaju i identifikaciji različitih elemenata unutar slike.

U trećoj glavi dat je pregled metoda za validaciju uspješnosti segmentacije, pri čemu su opisane njihove osnovne osobine. Takođe, objašnjeno je kako svaki od pristupa doprinosi procjeni rezultata segmentacije.

U četvrtoj glavi, predstavljeni su rezultati eksperimentalne analize uticaja kolornih modela pri segmentaciji slike uz pomoć algoritama. U glavi je dat detaljan i sistematičan opis datasetova.

U posljednjoj glavi, rezimirani su prezentovani rezultati, a predstavljeni su i identifikovani pravci budućih istraživanja u oblasti segmentacije digitalnih slika i kolornih modela.

# 1. SEGMENTACIJA SLIKE

Segmentacija slike je jedan od ključnih problema u oblasti kompjuterske vizije, podržan znatnim brojem sprovedenih istraživanja koja uključuju i algoritme zasnovane na obradi slike i tehnike zasnovane na učenju.

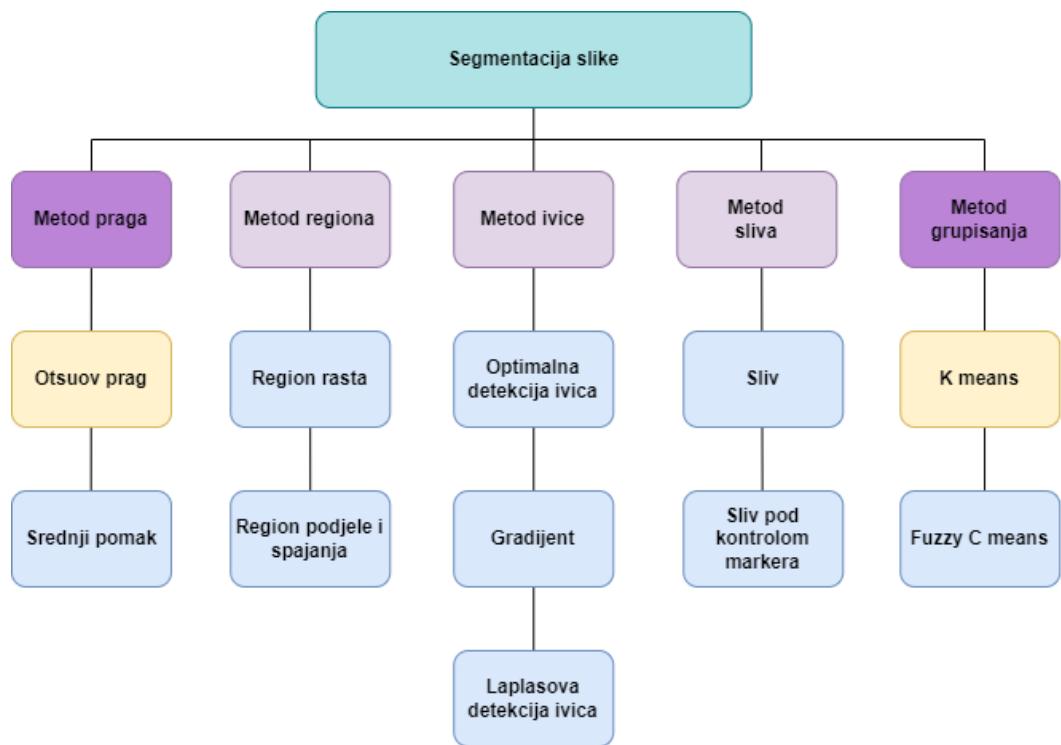
Zasnovana na principima digitalne obrade slike, segmentacija slike ima za cilj grupisanje sličnih regiona ili segmenata slike pod njihovim odgovarajućim oznakama klasa. S obzirom da je cijeli proces digitalan, analogne slike su predstavljene u obliku digitalne matrice piksela. Ova matrica, koja čini digitalnu sliku, sastoji se od redova i kolona gde svaki element matrice, piksel, nosi informacije o osvjetljenosti i boji. Piksel je osnovna jedinica informacije u digitalnoj slici i predstavlja najmanji mogući prikaz detalja na slici. Vrijednosti koje se nalaze unutar ove matrice piksela su numeričke i mogu varirati zavisno od tipa slike. Kolor modeli predstavljaju metode kojima se specificira i prikazuje boja u digitalnim slikama. Osvjetljenje unutar matrice piksela odnosi se na svjetlost ili tamu svakog pojedinačnog piksela, i često je direktno povezano sa vrijednostima kanala u kolor modelima. Segmentacija slike je složen proces klasifikacije slike sa modelom koji precizno ukazuje na to gdje je odgovarajući objekat prisutan. (slika 1) [13].



*Slika 1. Lijeva kolona sadrži originalne slike, a desna slike koje predstavljaju rezultat segmentacije.*

## 1.1 Tehnike segmentacije slike

Segmentacija slike je prvo bitno započela od digitalne obrade slike u kombinaciji sa algoritmima za optimizaciju. Ovi algoritmi su koristili metode kao što su *region growing* i *snake algorithm*. Kod metode *region growing*, algoritam počinje sa inicijalno definisanim regijama i zatim progresivno proširuje te regije upoređujući vrijednosti piksela susjednih područja kako bi formirao koherentne segmente slike. S druge strane, *snake algorithm* koristi dinamički kontur model za detekciju objekata, gdje se inicijalna kriva, ili *snake*, postepeno pomicaju prema granicama objekta na slici prateći gradijente intenziteta piksela. Ove metode segmentacije su se primarno oslanjale na lokalne karakteristike slike, analizirajući i fokusirajući se na male razlike u intenzitetu i boji piksela unutar definisanih područja. Algoritmi koji su imali globalni pogled na ulaznu sliku došli su mnogo kasnije, kao što su adaptivni pragovi, Otsuov metod i algoritmi za grupisanje koji su predloženi među klasičnim metodama obrade slike (*K-means*) (slika 2) [13].



Slika 2. Klasifikacija tehnik za segmentaciju slike

## 1.2 Metod praga

U digitalnoj obradi slike, metod zasnovan na postavljanju praga predstavlja najjednostavniji metod segmentacije slika. Ovaj metod se često koristi za kreiranje binarnih slika iz sivoskaliranih slika, gdje se prag primjenjuje za razlikovanje objekata od pozadine na osnovu intenziteta svjetlosti. Sivoskalirana slika se sastoji od piksela koji prikazuju vrijednosti intenziteta svjetlosti koje variraju od 0 (potpuna crna) do 255 (potpuna bijela), idealno za analizu bez distrakcije boja. Nasuprot njima, kolorne slike koriste tri kanala boja—crveni (R), zeleni (G), i plavi (B)—svaki s vrijednostima od 0 do 255, čineći ih pogodnim za primjene gdje su boje ključne za interpretaciju vizuelnog sadržaja. [14].

### Definicija

Osnovne tehnike postavljanja praga uključuju transformaciju svakog piksela na slici u crni piksel ako je njegov intenzitet niži od unaprijed određene vrijednosti, poznate kao prag. Pikseli čiji je intenzitet veći od praga transformišu se u bijele piksele, što rezultira binarnom slikom gdje su objekti i pozadina jasno razdvojeni. Na primjeru (slika 3), ovaj proces dovodi do transformacije tornja u potpuno crnu boju, dok nebo postaje skroz bijelo.



*Slika 3. Ilustracija jednostavnog metoda segmentacije slike zasnovanog na postavljanju praga za selekciju pripadnosti piksela jednoj od dvije moguće klase*

## Automatsko postavljanje praga

Iako u nekim slučajevima, prag  $T$  može odabrati korisnik, postoji mnogo slučajeva u kojima korisnik želi da se prag automatski postavi pomoću nekog algoritma. U tim slučajevima, "najbolji" prag treba da bude automatski postavljen, s ciljem da se pikseli podijele što tačnije u dvije klase (npr. pikseli ispod praga treba da odgovaraju pozadini, a oni iznad nekim objektima od interesa na slici). Jedna od najpoznatijih i često korišćenih metoda za automatsko postavljanje praga je Otsuova metoda [15].

### 1.2.1 Otsuov metod

U domenu kompjuterske vizije i obrade slika, Otsuova metoda, nazvana po Nobuyuki Otsu, koristi se za automatsko određivanje praga sivoskalirane slike. U osnovnom obliku, ovaj algoritam daje jedan prag intenziteta koji se koristi za razdvajanje piksela u dvije klase: prvi plan (objekti) i pozadinu. Ovaj prag se određuje minimizacijom varijacije intenziteta unutar svake klase, ili ekvivalentno, maksimizacijom varijacije između klasa [16].

#### Formulacija

Neka je slika predstavljena korišćenjem različitih nivoa sive boje (od 1 do  $L$ ). Broj piksela na svakom od tih nivoa označavamo sa  $n_i$ , a ukupan broj piksela na slici označavamo sa  $N$ , što može biti izraženo kao  $N = n_1 + n_2 + \dots + n_L$ . Da bismo pojednostavili ovu diskusiju, pretpostavljamo da je histogram sivih nivoa normalizovan, što znači da se može smatrati distribucijom vjerovatnoće, odnosno aproksimacijom distribucije vjerovatnoće:

$$p_i = \frac{n_i}{N}, \quad p_i \geq 0, \quad \sum_{i=1}^L p_i = 1. \quad (1.1)$$

Korišćenjem praga nivoa  $k$ , pikseli se dijele u dvije klase: pozadina  $C_0$  (koja obuhvata piksele sa nivoima od 1 do  $k$ ) i objekti  $C_1$  (koja obuhvata piksele sa nivoima od  $k + 1$  do  $L$ ). Tada su, vjerovatnoća pojave klasa i srednji nivo klase dati sa:

$$\omega_0 = \Pr(C_0) = \sum_{i=1}^k p_i = \omega(k), \quad (1.2)$$

$$\omega_1 = \Pr(C_1) = \sum_{i=k+1}^L p_i = 1 - \omega(k) \quad (1.3)$$

i

$$\mu_0 = \sum_{i=1}^k i \Pr(i \mid C_0) = \sum_{i=1}^k i \frac{p_i}{\omega_0} = \frac{\mu(k)}{\omega(k)}, \quad (1.4)$$

$$\mu_1 = \sum_{i=k+1}^L i \Pr(i \mid C_1) = \sum_{i=k+1}^L i \frac{p_i}{\omega_0} = \frac{\mu_T - \mu(k)}{1 - \omega(k)} \quad (1.5)$$

gdje su:

$$\omega(k) = \sum_{i=1}^k p_i \quad (1.6)$$

$$\mu(k) = \sum_{i=1}^k i p_i \quad (1.7)$$

kumulativni momenti nultog i prvog reda histograma do  $K$ -toga nivoa, respektivno, a:

$$\mu_T = \mu(L) = \sum_{i=1}^L i p_i \quad (1.8)$$

je ukupni srednji nivo originalne slike. Sljedeća relacija se lako može provjeriti za bilo koji izbor praga  $k$ :

$$\omega_0 \mu_0 + \omega_1 \mu_1 = \mu_T, \quad \omega_0 + \omega_1 = 1. \quad (1.9)$$

Varijanse klase su date sa:

$$\sigma_0^2 = \sum_{i=1}^k (i - \mu_0)^2 \Pr(i | C_0) = \sum_{i=1}^k (i - \mu_0)^2 \frac{p_i}{\omega_0} \quad (1.10)$$

i

$$\sigma_1^2 = \sum_{i=k+1}^L (i - \mu_1)^2 \Pr(i | C_1) = \sum_{i=k+1}^L (i - \mu_1)^2 \frac{p_i}{\omega_0}. \quad (1.11)$$

Da bismo ocijenili "dobrotu" praga (na nivou  $k$ ) i njegovu sposobnost da razdvoji klase, uvode se sljedeće mjere diskriminantnog kriterijuma (ili mjere odvajanja klasa), koje se koriste u diskriminantnoj analizi:

$$\lambda = \frac{\sigma_B^2}{\sigma_W^2}, \quad \kappa = \frac{\sigma_T^2}{\sigma_W^2}, \quad \eta = \sigma_B^2 / \sigma_T^2 \quad (1.12)$$

gdje su:

$$\sigma_W^2 = \omega_0 \sigma_0^2 + \omega_1 \sigma_1^2 \quad (1.13)$$

i

$$\sigma_B^2 = \omega_0 (\mu_0 - \mu_T)^2 + \omega_1 (\mu_1 - \mu_T)^2 = \omega_0 \omega_1 (\mu_1 - \mu_0)^2, \quad (1.14)$$

zbog jednačine (9) i

$$\sigma_T^2 = \sum_{i=1}^L (i - \mu_T)^2 p_i, \quad (1.15)$$

varijanse između klase i ukupna varijansa nivoa, respektivno. Ovaj problem se svodi na problem optimizacije u kojem treba pronaći prag  $k$  koji maksimizuje određenu funkciju cilja, kao što je  $\lambda$ ,  $\kappa$  ili  $\eta$ , koje su definisane sa (1.12). Ovo se temelji na pretpostavci da će dobar prag rezultirati boljim razdvajanjem klase na sive boje, dok će prag koji daje najbolje razdvajanje biti najbolji prag.

Važno je napomenuti da su diskriminativni kriterijumi koji se koriste za maksimizaciju  $\lambda$ ,  $k$  i  $\eta$  ekvivalentni jedan drugom. Na primjer, možemo reći da su  $k = \lambda + 1$  i  $\eta = \lambda/(\lambda + 1)$  jer sljedeća osnovna relacija uvijek važi:

$$\sigma_W^2 + \sigma_B^2 = \sigma_T^2. \quad (1.16)$$

Ovo je osnovna relacija koja povezuje različite diskriminativne kriterijume, i oni su ekvivalentni kada je riječ o optimizaciji problema. Stoga, problem pronalaženja praga  $k$  koji maksimizira bilo koji od ovih kriterijuma svodi se na isti optimizacioni problem.

Primjećuje se da su  $\sigma_W^2$  i  $\sigma_B^2$  funkcije graničnog nivoa  $k$ , ali je  $\sigma_T^2$  nezavisan od  $k$ . Takođe se napominje da se  $\sigma_W^2$  zasniva na statistici drugog reda (varijanse klase), dok se  $\sigma_B^2$  bazira na statistici prvog reda (srednja vrijednost klase). Stoga je  $\eta$  najjednostavnija mjera u odnosu na  $k$  i  $\eta$  se usvaja kao kriterijumska mjera za procjenu "dobrote" praga na nivou  $k$ .

Optimalni prag  $k^*$  koji maksimizira  $\eta$ , ili ekvivalentno maksimizira  $\sigma_B^2$ , bira se u sljedećem sekvenčijalnom pretraživanju korišćenjem jednostavnih kumulativnih veličina jednačina (6) i (7), ili eksplicitno koristeći jednačine od (1.2) do (1.5):

$$\eta(k) = \sigma_B^2(k)/\sigma_T^2 \quad (1.17)$$

$$\sigma_B^2(k) = \frac{[\mu_T \omega(k) - \mu(k)]^2}{\omega(k)[1 - \omega(k)]} \quad (1.18)$$

a optimalni prag  $k^*$  je:

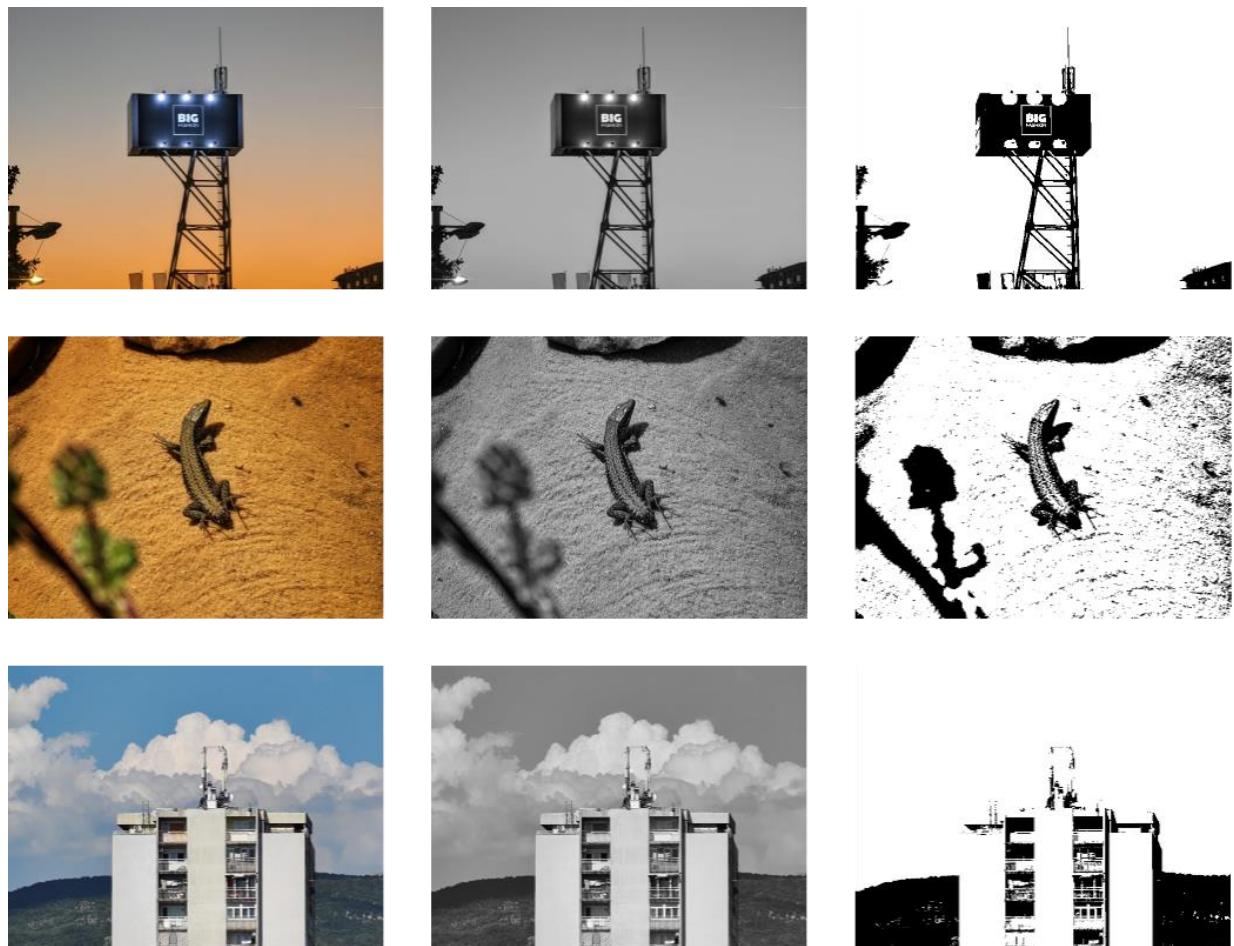
$$\sigma_B^2(k^*) = \max_{1 \leq k < L} \sigma_B^2(k) \quad (1.19)$$

Iz problema se opseg  $k$  u kojem se traži maksimum može ograničiti na:

$$S^* = \{k; \omega_0 \omega_1 = \omega(k)[1 - \omega(k)] > 0, \\ 0 < \omega(k) < 1\} \quad (1.20)$$

Nazvaćemo ga efektivnim opsegom histograma nivoa sive [16].

Opisani pristup se primjenjuje na sivoskaliranim slikama. To znači da slika mora biti u obliku crno-bijelog (*grayscale*) kanala prije nego što se primijeni Otsuova metoda (slika 4).



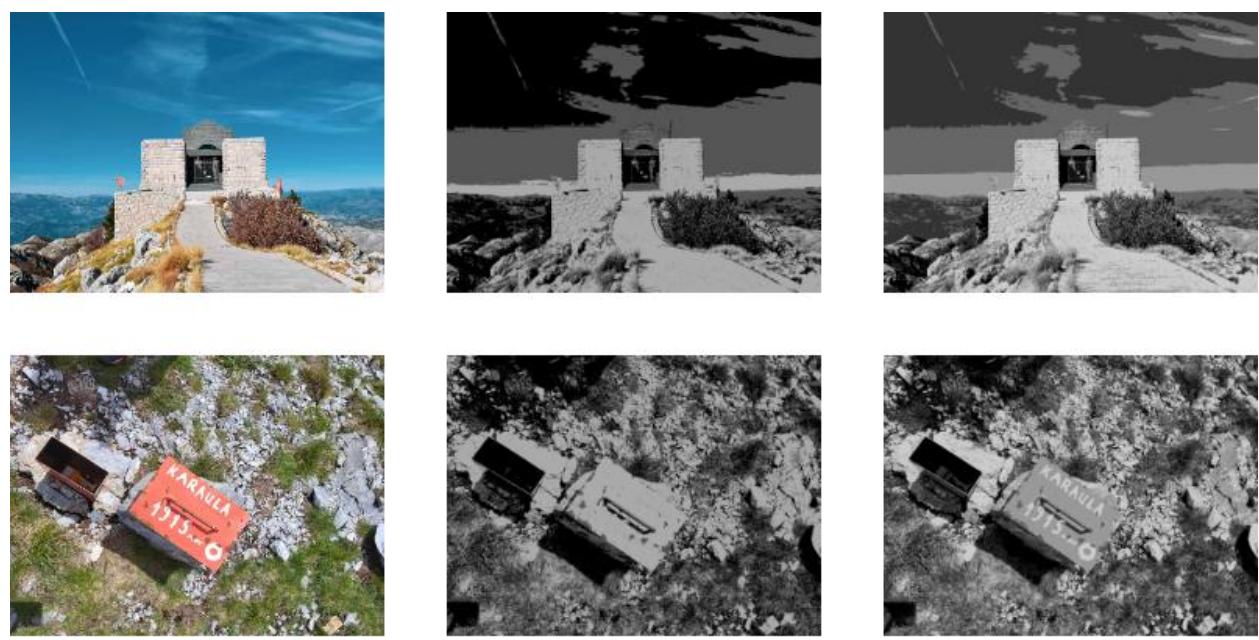
*Slika 4. Primjena Otsuovog metoda u segmentaciji digitalnih slika. Prva kolona – kolor slika, druga kolona – sivoskalirana slika dobijena na osnovu originalne slike u boji, treća kolona – rezultat segmentacije primjenom Otsuovog metoda*

### 1.2.1.1 Otsuov metod sa višestrukim pragovima

Otsuov metod sa višestrukim pragovima (slika 5) je tehnika za segmentaciju slike koja se koristi za razdvajanje piksela ulazne slike u više različitih klasa na osnovu intenziteta sivih tonova. Tradicionalno Otsuov metod pronalazi jedan optimalan prag koji razdvaja piksele u dvije klase: one ispod i one iznad tog praga, dok Otsuov metod sa višestrukim pragovima proširuje ovaj pristup kako bi se omogućilo razdvajanje na više od dvije klase.

Otsuov metod sa višestrukim pragovima funkcioniše tako što prvo analizira histogram slike, koji predstavlja raspodjelu intenziteta sivih tonova. Bira se broj željenih klasa ( $n$ ), a algoritam zatim traži  $(n - 1)$  praga koji će najbolje razdvojiti histogram u te klase. Algoritam koristi kriterijum unutar-klasne varijanse kako bi pronašao optimalne pragove. Cilj je minimizirati varijansu unutar svake klase dok se maksimizira varijansa između klasa, što znači da su klase što je moguće homogenije, dok su razlike između njih što je moguće veće. Nakon određivanja optimalnih praga, svaki piksel slike se razvrstava u odgovarajuću klasu na osnovu svog intenziteta [26]. U Python-u funkcija za Otsuov metod sa višestrukim pragovima izgleda ovako:

```
thresholds = threshold_multotsu(gray_image, classes=num_classes)
```



Slika 5. Prva kolona: originalne slike, druga kolona – slike segmentirane u 3 klase, treća kolona – slike segmentirane u 5 klase

## 1.3 Metod grupisanja - klasterizacija

Klasterizaciju su 1932. godine uveli Nj. E. Driver i A. L. Kroeber u svom radu o "Kvantitativnom izražavanju kulturnog odnosa". Od tada je ova tehnika napravila veliki skok i koristi se za otkrivanje nepoznatog u brojnim područjima primjene. Klasterizacija, odnosno grupisanje, je vrsta nenadgledanog učenja gdje se pripadnost grupama, odnosno, klasama mora izvući iz neoznačenih skupova podataka. Generalno, koristi se za hvatanje smislene strukture, osnovnih procesa i grupisanja svojstvenih skupu podataka. U grupisanju, zadatak je podijeliti populaciju u nekoliko grupa na način da su podaci u istim grupama sličniji jedni drugima od podataka u drugim grupama. Ukratko, to je kolekcija objekata zasnovanih na njihovim sličnostima i razlikama.

Iako ne postoje posebni kriterijumi za dobro grupisanje i potpuno zavisi od korisnika kako ga želi koristiti za svoje specifične potrebe. Može se koristiti za pronalaženje neuobičajenih tačaka podataka/odvojenih podataka u podacima ili za identifikaciju nepoznatih svojstava kako bi se pronašlo odgovarajuće grupisanje u skupu podataka [17].

Segmentacija zasnovana na grupisanju je tehnika koja se koristi u segmentaciji slike kako bi se slika podijelila na različite regije ili objekte na osnovu koncepta grupisanja. To uključuje grupisanje piksela ili elemenata slike u klastere, gdje su elementi unutar istog klastera sličniji jedni drugima nego onima u drugim klasterima. Cilj je identifikovati regije na slici koje imaju neke zajedničke karakteristike ili atributе, što olakšava analizu i obradu specifičnih regija od interesa [13].

Postoji nekoliko tipova metoda segmentacije zasnovane na grupisanju, od kojih svaka koristi različite tehnike ili algoritme grupisanja. Među najpopularnijim je *K-means*.

## 1.3.1 K-means

*K-means* je algoritam kod kojeg se izračunava udaljenost između svake tačke podataka i centroida da bismo je dodijelili klasteru. Centroidi su centralne tačke koje predstavljaju svaki klaster u skupu podataka.

*K-means* klasterizacija je metoda vektorske kvantizacije, koja potiče iz obrade signala, koja ima za cilj da podijeli  $n$  zapažanja u  $k$  klastera u kojima svako zapažanje pripada klasteru s najbližom srednjom vrijednosti, služeći kao prototip klastera [18]. To je iterativni proces dodjeljivanja svake tačke grupama i polako se tačke podataka grupišu na osnovu sličnih karakteristika. Cilj je da se minimizira zbir udaljenosti između tačaka podataka i centra klastera, da se identificuje tačna grupa kojoj svaka tačka podataka treba da pripada.

### 1.3.1.1 Funkcionalni principi *K-means* algoritma

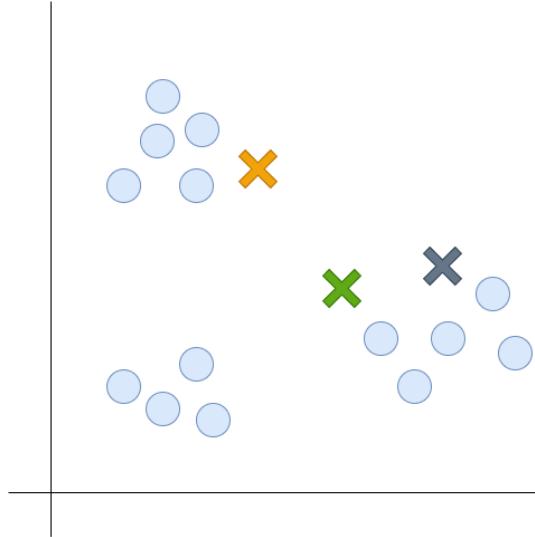
Algoritam prolazi kroz 5 koraka:

#### 1. Odabir broja klastera

Prvi korak je definisanje  $K$  broja klastera u koje će se grupisati podaci, izabratemo  $K = 3$ . Klasteri u *K-means* algoritmu su definisani kroz iterativni proces gdje se tačke podataka grupišu na osnovu njihove najmanje udaljenosti od centroida klastera. Na početku, centroidi su obično inicijalizovani na slučajno izabrane tačke podataka. U svakoj iteraciji, tačke podataka se pridružuju klasteru čiji je centroid najbliže njima, a zatim se centroidi ažuriraju.

#### 2. Inicijalizacija centara

Centroid je centar klastera, ali u početku će tačan centar tačaka podataka biti nepoznat, pa se biraju slučajne tačke podataka i definišu se kao centar za svaki klaster. U nastavku inicijalizuju se tri centroida u skupu podataka. Navedena inicijalizacija je ilustrovana na slici 6 (lijevo) za slučaj podataka u dvodimenzionom prostoru.



Slika 6. Ilustracija centroida na početku K-means algoritma

### 3. Dodjela tačke podataka najbližem klasteru

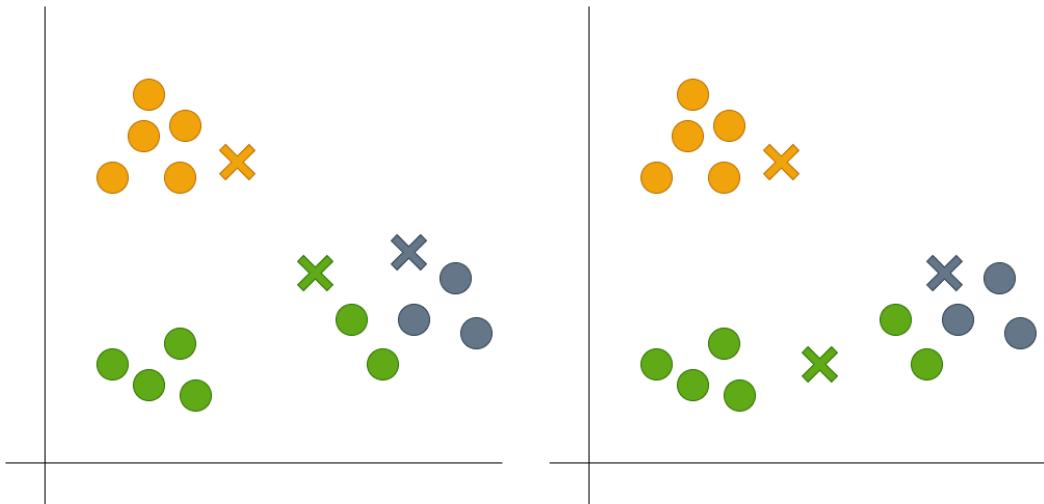
Kada su centroidi inicijalizovani, sljedeći korak je dodjeljivanje tačaka podataka  $X_n$  njihovom najbližem centroidu klastera  $C_k$ .

U ovom koraku prvo će se izračunati udaljenost između tačke podataka  $X_n$  i težišta  $C_k$  koristeći metriku Euklidske udaljenosti.

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (1.21)$$

Euklidska udaljenost, označena kao  $d(x, y)$ , predstavlja mjeru udaljenosti između dvije tačke  $x$  i  $y$  u  $n$ -dimenzionalnom prostoru. Ova udaljenost računa se kao kvadratni korijen sume kvadriranih razlika između odgovarajućih komponenti tačaka  $x$  i  $y$ , gdje  $x_i$  i  $y_i$  predstavljaju koordinate tačaka  $x$  i  $y$  u dimenziji  $i$ .

Zatim se bira klaster za date tačke koje imaju najmanje rastojanja, kao što je ilustrovano na slici 7.



Slika 7. Odabir klastera po najmanjem rastojanju (lijevo) i centralizovanje centroida (desno)

#### 4. Ponovo inicijalizovanje centroida

Kako bi se ponovno inicijalizovali centroidi, izračunava se prosječna vrijednost svih tačaka podataka unutar svakog klastera. Ovaj proces podrazumijeva izračunavanje nove pozicije centroida za svaki klaster na osnovu trenutnih članova klastera:

$$C_i = \frac{1}{|N_i|} \sum_{j=1}^{N_i} x_{ij} \quad (1.22)$$

$C_i$  je novi centroid i-tog klastera.

$|N_i|$  označava broj elemenata u i-toj grupi (tj. veličinu skupa podataka  $N_i$ ).

$x_{ij}$  predstavlja j-ti element unutar i-tog klastera

Suma  $\sum_{j=1}^{N_i} x_{ij}$  predstavlja zbir svih tačaka podataka unutar klastera  $i$  gdje  $j$  označava indeks tačke podataka unutar klastera

#### 5. Ponavljanje koraka 3 i 4

Koraci 3 i 4 će se ponavljati dok se ne dobije optimalan centar koji je ilustrovan na slici 8. Optimalan centar se postiže kada su pozicije centroida stabilizovane, odnosno kada se promjene između uzastopnih iteracija minimiziraju ili potpuno prestanu. To znači da dodatne iteracije neće

rezultirati značajnijim promjenama u raspodjeli tačaka po klasterima. Kada se dostigne ovo stanje, to takođe znači da se algoritam prekida, jer su postignuti optimalni uslovi za krajnje klasterisanje.



Slika 8. Optimalan centar

### 1.3.2 Segmentacija slike pomoću *K-means*

Razumijevanje i implementacija segmentacije slike pomoću *K-means* algoritma u Pythonu započinje učitavanjem slike kroz biblioteke kao što su *OpenCV*, koje slike učitavaju u BGR (Blue-Green-Red) formatu. Međutim, za prikazivanje slika, biblioteka *matplotlib* koristi RGB format, zbog čega je neophodna konverzija slike iz BGR u RGB format. Ova konverzija se vrši koristeći funkciju *cvtColor* iz *OpenCV* biblioteke, pripremajući sliku za dalju obradu.

Da bi se primijenio *K-means* algoritam, originalna trodimenzionalna slika, koja uključuje visinu, širinu i boje, mora se transformisati u dvodimenzionalni niz. Ovaj proces se ostvaruje korišćenjem funkcije *reshape* iz *numpy* biblioteke, kojom se svaki piksel pretvara u jedan red u nizu, dok kolone odgovaraju bojama tog piksela.

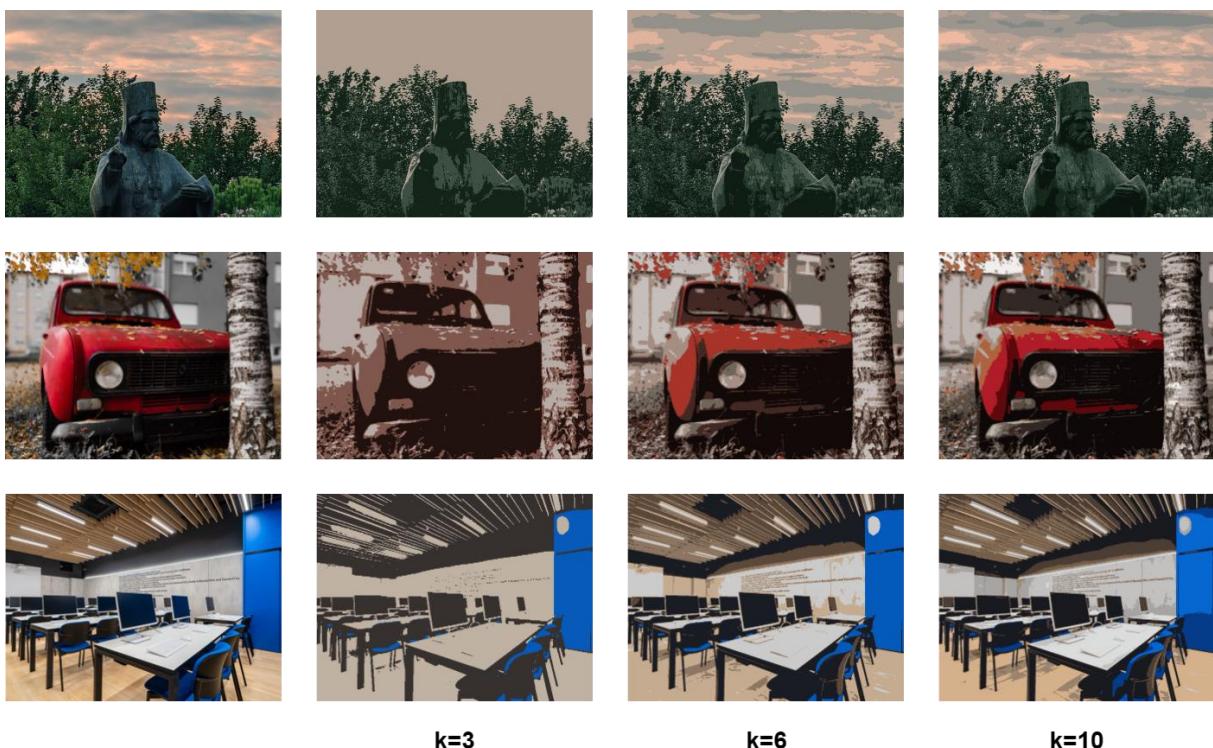
Svaki piksel u slikama tretira se kao tačka u prostoru čija je dimenzija definisana brojem boja ili kanala. *K-means* algoritam grupiše te piksele u klasterima bazirajući se na sličnosti njihovih boja, koristeći euklidsku udaljenost kao mjeru sličnosti. Svaki klaster karakteriše centroid, koji je definisan kao prosječna boja piksela unutar tog klastera. Kroz iterativni proces, pikseli se preraspoređuju klasterima sve dok se ne postigne optimalno grupisanje, čime se minimizira unutrašnja varijansa svakog klastera.

Nakon pripreme podataka, primjenjuje se sam *K-means* algoritam, gdje se broj klastera, označen kao  $k$ , može varirati u zavisnosti od analitičkih potreba. Algoritam zahtijeva jasno definisane kriterijume zaustavljanja kako bi se izbjeglo beskonačno izvršavanje. Obično se zaustavlja nakon određenog broja iteracija ili kada promjene u pozicijama centroida postanu zanemarljive. Ovo ukazuje na to da su klasteri stabilni i da je postignuta zadovoljavajuća preciznost grupisanja.

Završetkom algoritma, boje svih piksela ažuriraju se na boju njihovog centroida klastera, rezultirajući segmentiranom slikom koja omogućava vizuelno razlikovanje različitih dijelova na osnovu boja.

### Primjer implementacije:

U eksperimentima sa *K-means* segmentacijom, standardna vrijednost  $k$  obično je 3. Međutim, testirana je i vrijednosti  $k = 6$  i  $k = 10$  kako bi se ispitao uticaj broja klastera na segmentaciju. Ovaj pristup demonstrira kako promjene u parametru  $k$  mogu značajno uticati na vizuelni rezultat segmentirane slike, što je ilustrovano kod slike 9.



*Slika 9. Prva kolona: originalne slike, druga kolona – slike segmentirane u  $K = 3$  nivoa, treća kolona – slike segmentirane u  $K = 6$  nivoa, četvrta kolona – slike segmentirane u  $K = 10$  nivoa*

## 2. KOLORNI MODELI

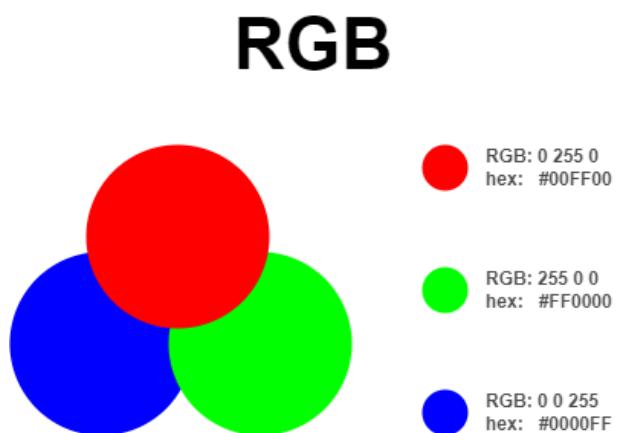
Kolorni modeli su sistemski načini predstavljanja i konceptualizacije boja pomoću numeričkih vrijednosti ili komponenata boje. Ovi modeli omogućavaju definisanje, analizu i reprodukciju boja u različitim kontekstima, kao što su umjetnost, dizajn, oglašavanje i mnogi drugi.

Svaki model boja dolazi sa jasno definisanim pravilima o tome kako se komponente tumače i kako se boje generišu ili reprodukuju. Kada se model boja poveže sa odgovarajućim pravilima i uslovima posmatranja, stvara se "prostor boja" koji opisuje načine na koje ljudsko oko percepira boje. Najpoznatiji kolorni modeli su:

- RGB (Red, Green, Blue),
- CMYK (Cyan, Magenta, Yellow, Key/Black),
- HSV (Hue, Saturation, Value).

### 2.1 RGB

Najrasprostranjeniji kolorni model je *RGB*, gdje je tačka boje u prostoru karakterizovana sa tri komponente boje odgovarajućeg piksela: crvena (*R*), zelena (*G*) i plava (*B*), kao što je ilustrovano na slikama 10 i 11 [19].



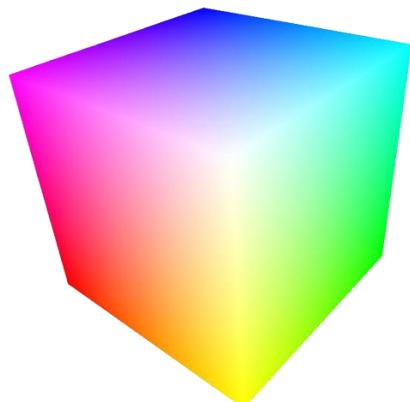
*Slika 10. Ilustracija RGB kolornog modela*



Slika 11. Originalna slika (lijevo) sa R, G i B komponentama (desno)

Tehnički, slike u RGB formatu predstavljaju se kao 3D matrice ili tenzori dimenzija  $V \times S \times 3$ , što znači da svaka slika ima tri sloja (jedan za svaku boju - R, G, B) koji zajedno formiraju kompletну sliku. Ova struktura omogućava detaljnu manipulaciju bojama i složeno analiziranje vizuelnih informacija.

RGB je model boja sa tri dimenzije – crvenom, zelenom i plavom – koje se miješaju kako bi se dobila određena boja. Na primjer, (255,0,0) predstavlja nijansu crvene boje, dok (0,0,255) predstavlja nijansu plave boje. Prilikom definisanja boja u ovim dimenzijama potrebno je poznavati redoslijed boja u spektru boja, npr. da kombinacija 100% crvene i zelene proizvodi žutu. RGB model boja se često geometrijski reprezentuje u obliku kocke mapiranjem crvenih, zelenih i plavih dimenzija na x, y i z osi u 3D prostoru. Ovo je ilustrovano na slici 12, gdje su sve moguće kombinacije boja predstavljene unutar granica kocke.

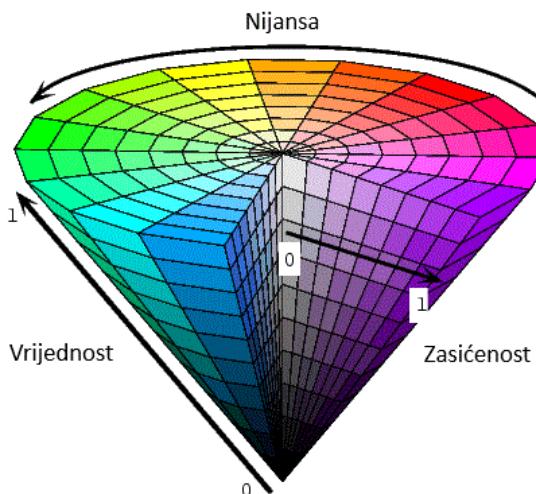


Slika 12. Kombinacije boja – RGB

RGB model se često koristi za analizu i segmentaciju slika na temelju njihovih boja, pri čemu se boje piksela koriste kao ključna karakteristika za razdvajanje različitih regija na slici.

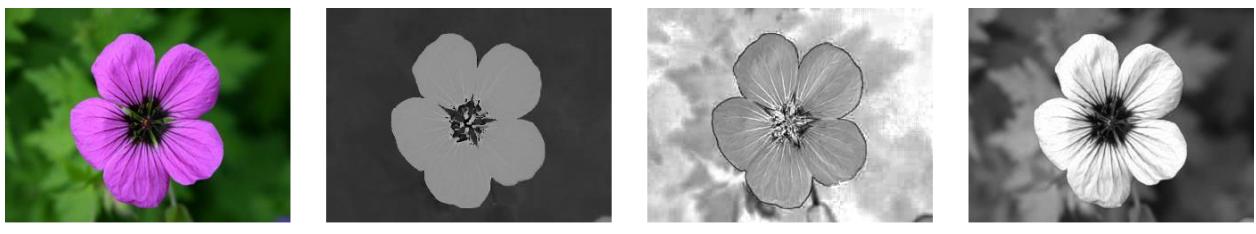
## 2.2 HSV

HSV (*engl. Hue, Saturation, Value*), ilustrovano na slikama 13 i 14 predstavlja prostor boja koji je znatno bliži RGB prostoru boja u kojem ljudi opisuju senzacije boja i percipiraju boje. Nijansa je dominantna boja koju posmatraju ljudi. Zasićenost je količina bijelog svjetla po izboru s nijansama. Vrijednost je intenzitet. Ukratko, Hue se odnosi na nijansu, Saturation se odnosi na zasićenost, a Value se odnosi na ton. Prostor boja HSV se može posmatrati kao geometrijski cilindar, gdje ugaona dimenzija predstavlja nijansu (H), počevši od primarne crvene na  $0^\circ$ , i kreće se do primarne zelene na  $120^\circ$  i primarno plave na  $240^\circ$ , a zatim se konačno vrati nazad u crveno na  $360^\circ$ . Udaljenost od centralne ose HSV cilindra odgovara Saturaciji (S). Kako se kreće prema vanjskoj ivici, vrijednost zasićenosti boje dostiže svoj maksimum, čime se postiže jasnija i definisanija nijansa. Centralna vertikalna ose prostora boja HSV je vrijednost (V), koja se kreće od crne na dnu sa svjetlinom ili vrijednošću 0, do bijele na vrhu sa svjetlinom ili vrijednošću 1 [20].



Slika 13. HSV kolorni model

Segmentacija slike u HSV prostoru podrazumijeva razdvajanje objekata na slici temeljem nijanse, zasićenosti i vrijednosti boje, što omogućuje precizniju izolaciju objekata različitih boja.



Hue - nijansa

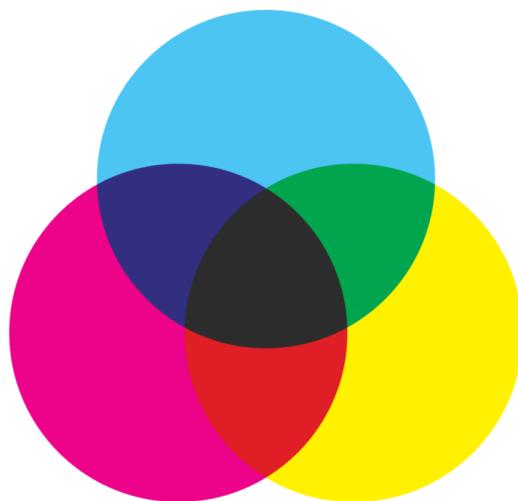
Saturation - zasićenost

Value - vrijednost

*Slika 14. HSV kolorni model i njegovi odgovarajući kanali*

## 2.3 CMYK

CMYK (slika 15) prostor boja uključuje subtraktivne primarne boje kao što su cijan (C), magenta (M), žuta (Y) i crna (K). Veliki raspon boja koje ljudi vide može se dobiti kombinovanjem cijan, magenta i žutih prozirnih mastila na bijeloj podlozi. Dodato je četvrto mastilo, crno kako bi se poboljšala reprodukcija nekih tamnih boja. Cijan boja upija crvenu svjetlost, ali propušta zelenu i plavu boju. Magenta boja apsorbuje zeleno svjetlo, ali prenosi crvenu i plavu boju, a žuta boja apsorbuje plavo svjetlo, ali prenosi crvenu i zelenu boju. Bijela boja reflektuje svjetlost koja se prenosi nazad do posmatrača. CMYK prostor boja se uglavnom koristi za štampače i za fotografije [20].



*Slika 15. CMYK kolorni model*

## 2.4 Konverzija i reprezentacija slike u kolornim modelima

### 2.4.1 Konverzija i reprezentacija slike u RGB modelu

Ako je piksel u BGR formatu predstavljen kao  $[B, G, R]$ , konverzija u RGB formatu se vrši zamjenom redoslijeda kanala:

Za piksel  $(i, j)$ :

$$BGR(i, j) = [B_{i,j}, G_{i,j}, R_{i,j}] \quad (2.1)$$

Konvertovani piksel u RGB formatu biće:

$$RGB(i, j) = [R_{i,j}, G_{i,j}, B_{i,j}] \quad (2.2)$$

Ova konverzija se primenjuje na svaki piksel slike.

. Ova konverzija se u Pythonu može obaviti jednom linijom koda koristeći *OpenCV* biblioteku:

```
image_rgb = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)
```

Funkcija *cv2.cvtColor* sa argumentima *cv2.COLOR\_BGR2RGB* automatski konverteuje BGR sliku u RGB format.

Otsuov metod kada radi sa RGB slikom, prvo dolazi do konvertovanja u sivoskaliranu. Otsuov metod zatim koristi histogram sivoskalirane slike da automatski odredi optimalni prag za segmentaciju. *K-means* algoritma kad radi sa RGB slikama koristi sve tri komponente za segmentaciju. Cilj je da podijeli piksele na osnovu njihove sličnosti u boji.

## 2.4.2 Konverzija i reprezentacija slike u HSV modelu

Prilikom konverzije slike iz BGR (plava, zelena, crvena) modela, koji je podrazumijevani model za *OpenCV* biblioteku, u HSV model, prolazi se kroz nekoliko koraka. Prvi korak naziva se normalizacija vrijednosti, gdje se vrijednosti B, G i R komponenti normalizuju na opseg od 0 do 1:

$$b_{norm} = \frac{b}{255.0}, \quad g_{norm} = \frac{g}{255.0}, \quad r_{norm} = \frac{r}{255.0} \quad (2.3)$$

Nakon toga, vrijednost (Value, V) se izračunava kao maksimalna vrijednost među normalizovanim vrijednostima:

$$V = \max(b_{norm}, g_{norm}, r_{norm}) \quad (2.4)$$

Delta se računa kao razlika između maksimalne i minimalne vrijednosti normalizovanih vrijednosti. Delta predstavlja razliku između najintenzivnije i najmanje intenzivne boje u pikselu. Ova vrijednost je ključna za određivanje zasićenja (Saturation, S) i nijanse (Hue, H). Kada je delta jednaka nuli, to znači da su sve tri komponente B, G, i R jednake, što rezultira nijansom koja je neodređena (obično se postavlja na 0) i nema zasićenja (što znači da je boja zapravo sivoskalirana).

$$\Delta = \max(b_{norm}, g_{norm}, r_{norm}) - \min(b_{norm}, g_{norm}, r_{norm}) \quad (2.5)$$

Nijansa (Hue) se računa na osnovu odnosa između normalizovanih vrijednosti. Izračunavanje zavisi od toga koja komponenta ima maksimalnu vrijednost:

Ako je  $\Delta = 0$ , tada je  $H=0$ .

Ako je  $\max(b_{norm}, g_{norm}, r_{norm}) = r_{norm}$ , tada je:

$$H = 60 \left( \frac{(g_{norm} - b_{norm})}{\Delta} \bmod 6 \right) \quad (2.6)$$

$\bmod 6$  se koristi da bi se osiguralo da vrijednosti nijanse ostane u opsegu od 0 do 360 stepeni.

Ako je  $\max(b_{norm}, g_{norm}, r_{norm}) = g$ , tada je:

$$H = 60 \left( \frac{(b_{norm} - r_{norm})}{\Delta} + 2 \right) \quad (2.7)$$

Dodaje se 2 da bi se vrijednost nijanse pomjerila u opseg od 120 do 180 stepeni.

Ako je  $\max(b_{norm}, g_{norm}, r_{norm}) = b_{norm}$ , tada je:

$$H = 60 \left( \frac{(r_{norm} - g_{norm})}{\Delta} + 4 \right) \quad (2.8)$$

Dodaje se 4 da bi se vrijednost nijanse pomjerila u opseg od 240 do 300 stepeni.

Nijansa (H) se skalira na opseg od 0 do 180:

$$H = \frac{H}{2} \quad (2.9)$$

Zasićenje (Saturation, S) se računa kao razlika između maksimalne i minimalne vrijednosti normalizovanih vrijednosti, podeljena sa maksimalnom vrijednošću:

$$S = \frac{\Delta}{\max(b_{norm}, g_{norm}, r_{norm})} \quad (2.10)$$

Zatim se vrijednosti za S i V skaliraju nazad u opseg vrijednosti koje koristi OpenCV (od 0 do 255):

$$\begin{aligned} S &= S * 255 \\ V &= V * 255 \end{aligned} \quad (2.11)$$

Na ovaj način, HSV vrijednosti se dobijaju i mogu se koristiti u daljoj obradi slike. Ova konverzija se u Pythonu može obaviti jednom linijom koda koristeći *OpenCV* biblioteku:

```
image_hsv = cv2.cvtColor(image, cv2.COLOR_BGR2HSV)
```

Funkcija `cv2.cvtColor` sa argumentima `cv2.COLOR_BGR2HSV` automatski konvertuje BGR sliku u HSV format, koristeći opisane matematičke korake.

Otsuov metod koristi V (Value) kanal kada se radi sa HSV slikom, jer V kanal predstavlja osvjetljenost piksela, što je ključno za pragovanje. Kada radimo sa HSV slikom, *K-means* koristi sve tri komponente (H, S, V) za klasterizaciju, jer one zajedno pružaju kompletну informaciju o boji i osvjetljenosti piksela.

### 2.4.3 Konverzija i reprezentacija slike u CMYK modelu

Da bi slika bila konvertovana iz BGR u CMYK format, prvo dolazi do konverzije slike iz BGR u RGB sliku. Konverzija iz RGB u CMYK je češći i bolji pristup jer su RGB i CMYK standardni modeli boja za digitalne prikaze i štampu, respektivno. Prvi korak je normalizacija vrijednosti boja iz opsega [0, 255] na opseg [0, 1] deljenjem svake komponente ( $R, G, B$ ) sa 255:

$$R_{norm} = \frac{R}{255}, \quad G_{norm} = \frac{G}{255}, \quad B_{norm} = \frac{B}{255} \quad (2.12)$$

Početne vrijednosti za C, M i Y komponente su zatim izračunate oduzimanjem normalizovanih vrijednosti crvene, zelene i plave komponente od 1:

$$C = 1 - R_{norm}, \quad M = 1 - G_{norm}, \quad Y = 1 - B_{norm} \quad (2.13)$$

Sledeći korak je određivanje vrijednosti K (black) kao minimalne vrijednosti između C, M i Y:

$$K = \min(C, M, Y) \quad (2.14)$$

Nakon toga, vrijednosti C, M i Y su prilagođene oduzimanjem K od svake od njih i dijeljenjem rezultata sa  $1 - K$ , pod uslovom da K nije jednak 1; ukoliko jeste, C, M i Y se postavljaju na 0:

$$C = \frac{C - K}{1 - K}, \quad (\text{ako } K \neq 1, \text{ inače } C = 0) \quad (2.15)$$

$$M = \frac{M - K}{1 - K}, \quad (\text{ako } K \neq 1, \text{ inače } M = 0) \quad (2.16)$$

$$Y = \frac{Y - K}{1 - K}, \quad (\text{ako } K \neq 1, \text{ inače } Y = 0) \quad (2.17)$$

Konačno, prilagođene vrijednosti C, M, Y i K su skalirane iz opsega [0, 1] na opseg [0, 100] množenjem sa 100, čime su dobijene konačne vrijednosti za CMYK komponente:

$$C_{scaled} = C * 100, \quad M_{scaled} = M * 100, \quad (2.18)$$

$$Y_{scaled} = Y * 100, \quad K_{scaled} = K * 100 \quad (2.19)$$

Ovaj postupak omogućava da slika bude konvertovana iz RGB formata, u CMYK format. Ova konverzija se u Pythonu može obaviti uz pomoć sljedećih koda:

```
def rgb_to_cmyk(image):
    CMYK_SCALE = 100
    BGR_SCALE = 255

    def convert_pixel(r, g, b):
        if (r, g, b) == (0, 0, 0):
            return 0, 0, 0, CMYK_SCALE
        c = 1 - r / BGR_SCALE
        m = 1 - g / BGR_SCALE
        y = 1 - b / BGR_SCALE
        min_cmy = min(c, m, y)
        c = (c - min_cmy) / (1 - min_cmy) if min_cmy != 1 else 0
        m = (m - min_cmy) / (1 - min_cmy) if min_cmy != 1 else 0
        y = (y - min_cmy) / (1 - min_cmy) if min_cmy != 1 else 0
        k = min_cmy
        return c * CMYK_SCALE, m * CMYK_SCALE, y * CMYK_SCALE, k * CMYK_SCALE
```

*K*-means algoritam koristi sve komponente CMYK modela kako bi grupisao slične piksele na osnovu njihove ukupne boje, dok Otsuv metod obično koristi samo jedan kanal (najčešće K kanal) za određivanje optimalnog praga binarizacije. Na taj način, *K*-means algoritam uzima u obzir kompleksne boje, dok Otsu metoda koristi informacije o tamnim i svijetlim djelovima slike za segmentaciju.

### 3. Metode validacije

Metode, odnosno metrike validacije se koriste da na adekvatan način opišu uspješnost segmentacije. Neke od poznatih metrika, koje su korišćene u eksperimentalnoj analizi prezentovani u ovom radu su: Koeficijent Dice, F-mjera, Jaccardov indeks i Varijacija informacija.

Prva metrika *Dice Coefficient* (Koeficijent Dice), mjeri sličnost između segmentirane slike i *ground-truth* slike, pružajući informacije o stepenu preklapanja. *Ground-truth* predstavlja precizno označene segmente na slikama koje služe kao tačan standard za evaluaciju i obučavanje algoritama segmentacije. Primjer *ground-truth*-a je prikazan na slici 16. Druga metrika *F-Measure* (F-Mjera) - harmonijska sredina preciznosti i opoziv, je korisna za balansiranje tačnosti i potpunosti segmentacije u odnosu na *ground-truth*. Treća metrika – *Jaccard Index* (Jaccardov Indeks) mjeri veličinu preklapanja između segmentirane i *ground-truth* slike u odnosu na njihovu uniju. Četvrta metrika Variation of Information (Varijacija informacija) kvantificira količinu informacija koja se gubi ili dobija prilikom segmentacije.



Slika 16. Primjer ground-truth-a

### 3.1 Dice koeficijent

*Dice* koeficijent (DSC) je metrika sličnosti koja se obično koristi u segmentaciji slika, obradi prirodnog jezika i drugim poljima gdje postoji potreba za mjerjenjem sličnosti između dva skupa. Ilustracija koncepta je data na slici 17. Navedena metrika je poznata i pod nazivom „*Sørensen-Dice koeficijent*“, prvi su ga uveli Lee R. Dice 1945. i Thorvald Sørensen 1948. godine [21] [22].

Koeficijent Dice je mjera sličnosti između dva skupa,  $A$  i  $B$ . Koeficijent se kreće od 0 do 1, gdje 1 označava da su dva skupa identična, a 0 da se dva seta ne preklapaju. Definiše se kao:

$$DSC = \frac{|A \cap B|}{(|A| + |B|)}, \quad (3.1)$$

gdje:

$|A|$  predstavlja broj elemenata (kardinalnost) u skupu A

$|B|$  predstavlja broj elemenata (kardinalnost) u skupu B

$|A \cap B|$  predstavlja broj elemenata koji su prisutni u oba skupa.

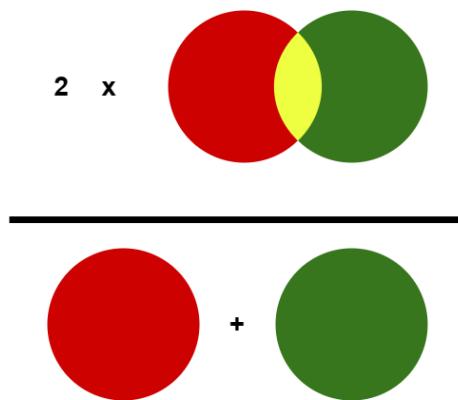
U nastavku će biti dat kratak primjer u cilju ilustracije razmatrane metrike. Razmatraju se dva skupa:

$$A = \{1, 2, 3, 5, 8, 9\}, \quad B = \{1, 3, 5, 7, 9\}$$

Presjek A i B je  $\{1, 3, 5, 9\}$ , a veličina A i B je 6 i 5, respektivno. Dakle, *Dice* koeficijent je:

$$DSC = \frac{|A \cap B|}{(|A| + |B|)} = 2 * \frac{4}{(6 + 5)} = 0.727$$

To znači da su skupovi A i B slični 72,7%.



Slika 17. Ilustracija formule za koeficijent Dice

Dice koeficijent, zbog svojih karakteristika, nudi nekoliko prednosti u odnosu na druge metrike sličnosti, posebno kada se radi o neuravnoteženim skupovima podataka gdje jedan skup može biti znatno veći od drugog. Ova metrika je posebno korisna za probleme segmentacije slika jer je osjetljivija na preklapanje između predviđene i osnovne (*ground-truth*) maske. Maski segmentacije pristupa se kao skupovima piksela, pri čemu su predviđena i osnovna segmentacija predstavljene binarnim maskama koje određuju da li je piksel dio segmentiranog objekta ili ne.

Visoke vrijednosti Dice koeficijenta ukazuju na visok nivo sličnosti između predviđene i osnovne maske, što implicira da model segmentacije ili algoritam postiže dobre performanse. Suprotno, niske vrijednosti ovog koeficijenta ukazuju na loše performanse segmentacije.

## 3.2 F-mjera

U kontekstu binarne klasifikacije analize, F-mjera (F) ili F-rezultat služi kao metrika tačnosti testa. Ova mjera se izračunava koristeći dvije komponente: preciznosti i opoziv. Preciznost se odnosi na udio istinito, odnosno stvarno pozitivnih rezultata u ukupnom broju pozitivnih rezultata testa, uključujući i one koji su pogrešno klasifikovani. Opoziv označava udio istinito pozitivnih rezultata među svim slučajevima koji bi trebalo da su prepoznati kao pozitivni.  $F_1$ -mjera predstavlja sredinu preciznosti i opoziva, efektivno simbolizirajući oba ova aspekta u jedinstvenoj metričkoj vrijednosti. Ilustracija koncepta je predstavljena na slici 18.

Maksimalna vrijednost F-mjere je 1.0, što predstavlja idealnu preciznost i opoziv, dok je minimalna moguća vrijednost 0, što se događa u slučajevima gdje je preciznost ili opoziv jednak nuli [23].

Preciznost i opoziv su definisani sljedećim izrazima:

$$P = \frac{TP}{TP + FP}, \quad (3.2)$$

$$O = \frac{TP}{TP + FN}. \quad (3.3)$$

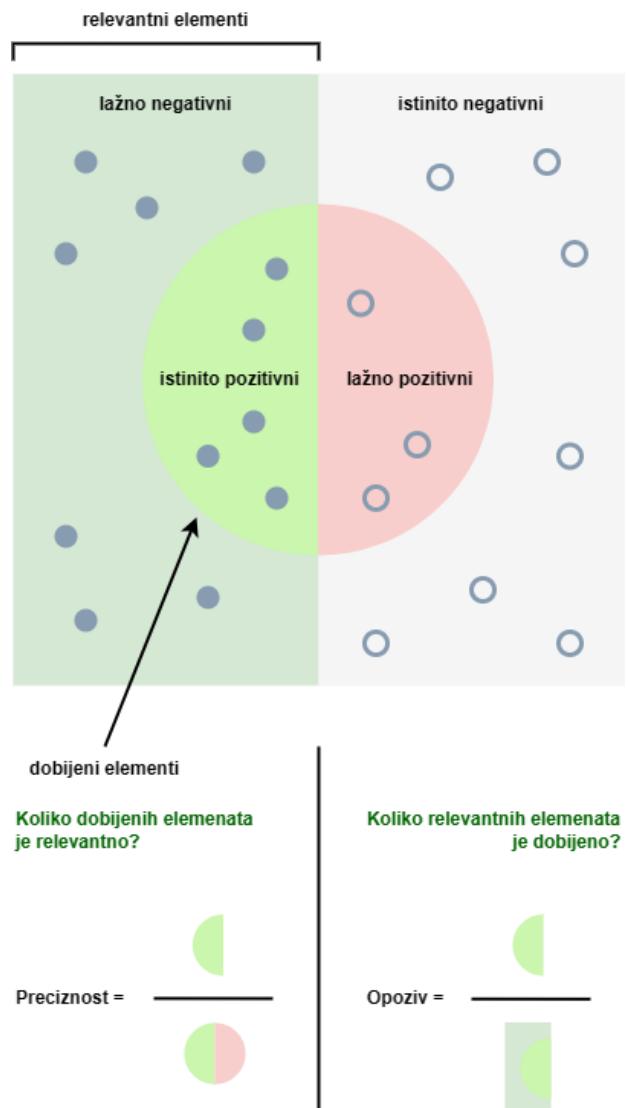
Na bazi gornjih definicija, F1-mjera se definiše na sljedeći način:

$$F_1 = \frac{2}{\frac{1}{P} + \frac{1}{O}} = \frac{2 * P * O}{P + O} = \frac{2TP}{2TP + FP + FN} \quad (3.4)$$

U nastavku će biti dat kratak primjer u cilju ilustracije razmatrane metrike. Razmatra se metod koji analizira slike automobila i autobusa. U ovom primjeru, metodi je predstavljena slika koja sadrži osam automobila i četiri autobusa, no model pogrešno identificuje samo četiri automobila, od kojih je jedan zapravo autobus. Ova greška pokazuje da je preciznost modela 3/4, jer su tri od četiri vozila koja je model prepoznao kao automobile zaista bili automobili. Što se tiče opoziva, model postiže rezultat od 3/8, s obzirom da je uspio identifikovati samo tri od osam stvarnih automobila na slici.

Stoga je F-Mjera:

$$F_1 = \frac{2PO}{P + O} = 2 \frac{\left(\frac{3}{4} * \frac{3}{8}\right)}{\left(\frac{3}{4} + \frac{3}{8}\right)} = 0.5$$



Slika 18. Ilustracija F-Mjera metrike

Kod segmentacije slike, preciznost se odnosi na to koliko su tačno segmenti identifikovani od strane algoritma u odnosu na stvarne, željene segmente (*ground-truth*). Opoziv mjeri koliko su efikasni svi relevantni segmenti uhvaćeni od strane algoritma. Visoka preciznost znači da je većina identifikovanih segmenata tačna, dok visoki opoziv znači da je većina stvarnih segmenata uspješno identifikovana. Dakle, F-mjera sintetizuje ova dva aspekta - preciznost i opoziv - pružajući jedinstvenu mjeru za ocjenu performansi algoritma segmentacije.

## 3.2 Jaccard-ov indeks

Jaccard-ov indeks (JI), takođe poznat kao Jaccard-ov koeficijent sličnosti, je statistička metrika koja se koristi za mjerjenje sličnosti i raznolikosti skupova uzoraka. Razvio ga je Grove Karl Gilbert 1884. kao njegov odnos provjere [24].

Jaccard-ov koeficijent mjeri sličnost između konačnih skupova uzoraka i definisan je kao odnos kardinalnosti presjeka dva skupa i kardinalnosti unije dva skupa. Kardinalnost je izraz koji koristimo za broj elemenata, odnosno veličinu skupa.

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}. \quad (3.5)$$

Ako dva skupa podataka dijeli potpuno iste članove onda će njihov Jaccard-ov indeks sličnosti biti 1, a ako nema zajedničkih članova onda će Jaccard-ov indeks sličnosti biti 0. Koncept je ilustrovan na slici 19.

U nastavku će biti dat kratak primjer u cilju ilustracije razmatrane metrike. Razmatraju se dva skupa:

$$A = \{1, 3, 5, 7, 9\}, \quad B = \{1, 2, 3, 4, 5, 6, 7, 8\}$$

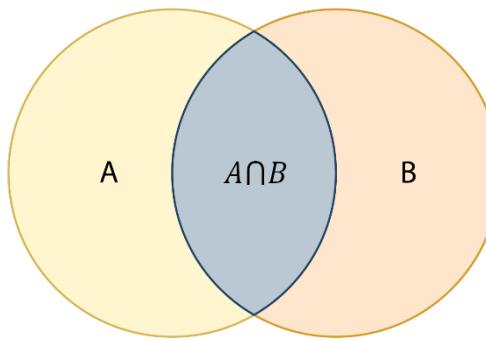
Da bismo izračunali Jaccard-ov indeks između njih, prvo brojimo ukupan broj elemenata u oba skupa, a zatim taj ukupan broj podijelimo s ukupnim brojem zapažanja u oba skupa. Skupovi su sljedećih dimenzija:

$$|A \cap B| = \{1, 2, 3, 5, 7, 6, 9\} = 7,$$

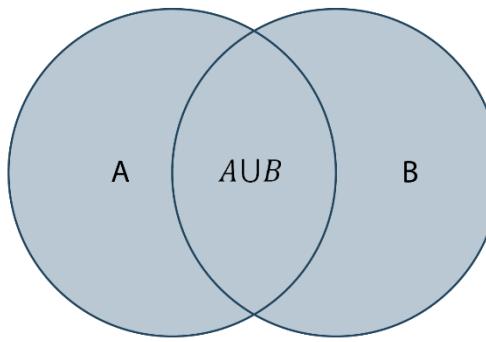
$$|A \cup B| = \{1, 2, 3, 4, 5, 6, 7, 8, 9\} = 9.$$

Jaccard-ov indeks je:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|7|}{|9|} = 0.78.$$



$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}$$



Slika 19. Ilustracija izračunavanja Jaccardovog indeksa

Jaccard-ov indeks procjenjuje sličnost i raznolikost između dva seta podataka, što u kontekstu segmentacije slike znači koliko su segmenti koje je identifikovao algoritam slični segmentima *ground-truth-a*. Vrijednost Jaccardovog indeksa varira od 0 do 1, gdje 0 označava da nema preklapanja između predviđenih segmenata i *ground-truth* segmenata (nema sličnosti), a 1 označava savršeno preklapanje. Jaccardov indeks je koristan u segmentaciji slika jer direktno mjeri kvalitet segmentacije bez obzira na veličinu segmenta, pružajući jasnu i direktnu mjeru tačnosti segmentacije.

### 3.4 Varijacija informacija

Varijacija informacija (VI) jeste metrika koja se koristi u teoriji vjerovatnoće i teoriji informacija za mjerjenje udaljenosti ili različitosti između dvije particije ili grupisanja elemenata. Koncept je ilustrovan na slici 20. Varijacija informacija se izračunava kao [25]:

$$VI(X, Y) = H(X) + H(Y) - 2 \times MI(X, Y). \quad (3.6)$$

Gdje je:

- $H(X)$  entropija koja predstavlja mjeru “nereda” unutar skupa X i izračunava se koristeći vjerovatnoću pojavljivanja svakog ishoda unutar skupa:

$$H(X) = - \sum_i p_i \log(p_i) \quad (3.7)$$

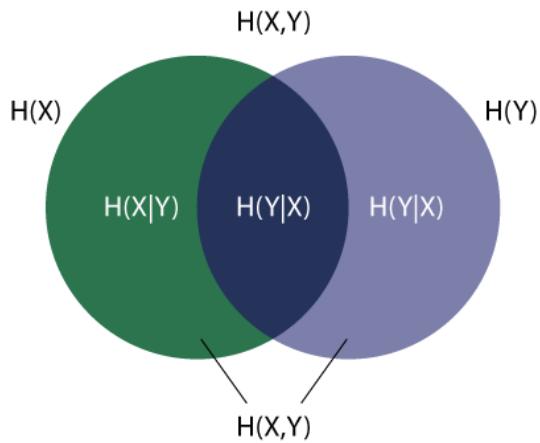
- $H(Y)$  entropija koja predstavlja mjeru “nereda” unutar skupa Y i izračunava se koristeći vjerovatnoću pojavljivanja svakog klastera unutar skupa:

$$H(Y) = - \sum_j q_j \log(q_j) \quad (3.8)$$

- $MI(X, Y)$  je mjera koliko informacija dva skupa dijele – međusobno informisanje:

$$MI(X, Y) = \sum_{i \in A, j \in B} P(i, j) \log \left( \frac{P(i, j)}{P(i)P(j)} \right) \quad (3.9)$$

Kod relacije (3.9)  $P(i,j)$  predstavlja združenu vjerovatnoću dva ishoda, vjerovatnoće pojedinačnih ishoda i i j..



Slika 20. Ilustracija metrike varijacije informacija

Za razliku od prethodnih metrika, kod varijacije infromacija vrijednost bliža nuli predstavlja dobar rezultat, dok vrijednost bliža jedinici predstavlja loš rezultat. Uobičajeno je da se dobijene vrijednosti varijacije informacija oduzimaju od 1.

Varijacija informacija se koristi za segmentaciju slika jer pruža mjeru disimilarnosti između dva skupa segmentacija, tj. rezultata segmentacije primjenom razmatranog algoritma i *ground-truth* segmentacije, odnosno željene segmentacije. Varijacija informacija kvantificira koliko su dva rezultata slična u pogledu informacija koje dijele.

## 4. Eksperimentalni rezultati

U okviru rezultata primjenom  $K$ -means algoritma i Otsuovog metoda, pažljivo će se analizirati kako odabir kolornog modela (RGB, HSV, CMYK) utiče na efikasnosti i preciznost segmentacije. Eksperimentisalo se sa  $K$ -means segmentacijom koristeći različite vrijednosti  $K$ , odnosno broja klastera, kako bi se razumio njihov uticaj na rezultate segmentacije za svaki kolorni model. Pored  $K$ -means segmentacije, primjenjivan je i Otsuov metod za automatsko određivanje praga segmentacije u kontekstu ovih kolornih modela. Ovaj metod je omogućio automatsko određivanje optimalnog praga za binarnu segmentaciju slike. Dodatno, ispitivan je i Otsuov metod sa višestrukim pragovima (Multi-Otsu) sa različitim vrijednostima klasa.

Evaluacija rezultata segmentacije primjenom ovih algoritama obavljena je primjenom sljedećih metrika: Koeficijent Dice, F-mjera, Jaccardov indeks i Varijacija informacija.

Eksperimentalna analiza u istraživanju obuhvata tri različita *dataset*-a, koji su korišćeni za testiranje efikasnosti i preciznosti segmentacijskih algoritama na različitim tipovima slika. Prvi *dataset*, koji se sastoji od 2000 slika, obrađen je bez modifikacija u kontrastu i osvjetljenju. U drugom *dataset*-u, takođe sa 2000 slika, adaptacija je izvršena tako što je faktor kontrasta povećan na 2, a intenzitet osvjetljenja na 1.5, čime se ispituje robustnost algoritama na promjene u osvjetljenju i kontrastu. Treći *dataset* specifično je podijeljen na dva podseta, sa po 640 slika; prvi podset prikazuje unutrašnjost kuće, dok drugi uključuje slike prirodnih okruženja, testirajući kako algoritmi segmentiraju pojedinačne objekte u različitim okruženjima. Sve slike u datasetovima su formata 400xX piksela, gde X predstavlja varijabilnu veličinu, što doprinosi kompleksnosti obrade. Za svaku sliku, pripadajući *ground-truth* podaci su dostupni, omogućavajući kvantitativnu evaluaciju rezultata segmentacije.

Kada se govori o datasetovima koji sadrže veliki broj slika, važno je naglasiti da se analiza temelji na srednjim vrijednostima iz cijelog skupa podataka. To znači da primjeri prikazani u radu služe samo kao ilustracije i ne odražavaju nužno specifične karakteristike svake pojedinačne slike unutar *dataset*-a. Takođe, iako će neki primjeri slika demonstrirati uspješnu segmentaciju prema svim metrikama, ipak treba naglasiti da navedeno ne znači da su sve slike unutar *dataset*-a obrađene s istim nivoom uspješnosti. Primjeri ilustruju optimalan ishod koji je moguć, ali varijacije u kvalitetu segmentacije su očekivane kroz cijele datasetove.

## 4.1 K-means

### 4.1.1 Rezultati uspješnosti segmentacije

U nastavku je detaljno prikazana analiza eksperimentalnih rezultata, koji su ostvareni korištenjem  $K$ -means algoritma u kombinaciji sa kolornim modelima: RGB, CMYK i HSV. Efikasnost segmentacije je evaluirana koristeći četiri različite metrike: Dice koeficijent, F-mjera, Jaccard-ov indeks i Varijacija informacija. Vrijednosti ovih metrika su dobijene usrednjavanjem rezultata po *dataset-u*. Eksperimentalni rezultati realizovani su na odabranom uzorku (*dataset-u*) od 2000 kolornih slika. Slike su obrađene bez promjena kontrasta i osvjetljenja, čime su zadržane u njihovom originalnom vizuelnom stanju. Dimenzije slika su 400xX piksela, gdje je X varijabilna veličina. Za svaku sliku dostupni su i pripadajući *ground-truth* podaci.

Tabele 1 i 2 i slika 21 detaljno prikazuju rezultate uspješnosti segmentacije slike za dva različita broja klastera:  $K=3$  i  $K=5$ . Na slikama 22 i 23 prikazane su segmentirane slike pri  $K = 3$  i  $K = 5$ , ilustrujući razlike u segmentaciji u zavisnosti od broja klastera. Slike predstavljaju reprezentativne uzorke iz *dataset-a*.

*Tabela 1: Rezultati uspješnosti segmentacije kod K-means algoritma pri  $K = 3$  na dataset-u originalnih slika*

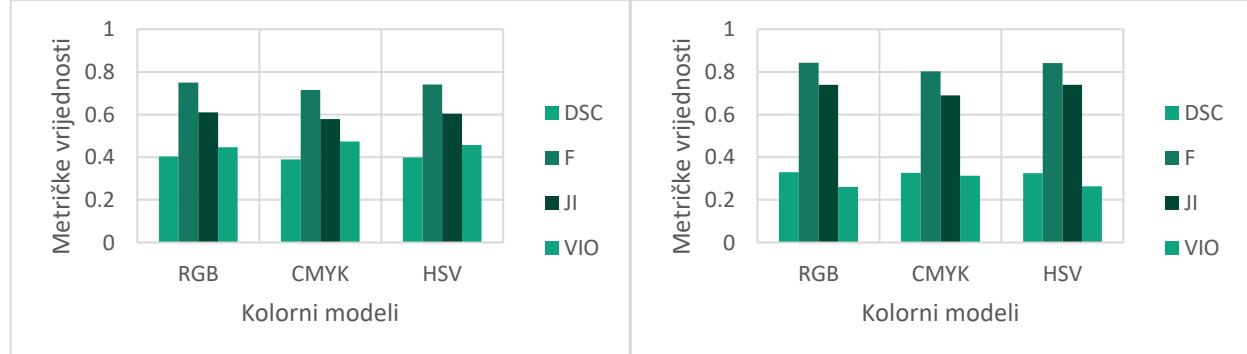
	DSC	F	JI	VI
RGB	<b>0.4036</b>	<b>0.7498</b>	<b>0.6106</b>	0.4467
CMYK	0.3900	0.7159	0.5782	<b>0.4732</b>
HSV	0.3987	0.7413	0.6043	0.4580

Tabela 2: Rezultati uspješnosti segmentacije kod K-means algoritma pri  $K = 5$  na dataset-u originalnih slika

	DSC	F	JI	VI
RGB	<b>0.3296</b>	<b>0.8430</b>	<b>0.7401</b>	0.2604
CMYK	0.3267	0.8019	0.6906	<b>0.3131</b>
HSV	0.3249	0.8421	0.7389	0.2633

U tabeli 1, gdje je primijenjen  $K$ -means algoritam sa  $K = 3$  na RGB slikama, DSC je relativno nizak, što implicira da segmentacija nije u potpunosti precizna. F-mjera je prilično visoka, što znači da postoji dobar balans između preciznosti i odziva. JI pokazuje solidnu uspješnost, što znači da ima pristojno preklapanje između segmentiranih i pravih regija. VI je srednje vrijednosti, što ukazuje na to da postoji prostor za poboljšanje u smislu informacijske konzistentnosti. Kod CMYK-a i HSV-a vidimo blagi rast kod VI, dok su vrijednosti ostalih metrika smanjene.

U tabeli 2, gdje je primijenjen  $K$ -means algoritam sa  $K=5$  dolazi do rasta F-mjere i Jaccard-ovog indeksa kod svih kolorni modela, dok je VI pao u odnosu na  $K=3$ .



Slika 21. Rezultati uspješnosti segmentacije kod K-means algoritma pri  $K = 3$  (lijevo) i pri  $K = 5$  (desno) na dataset-u originalnih slika



**RGB**



**CMYK**



**HSV**



*Slika 22. Segmentirane slike pomoću kolornih modela i K-means algoritma pri  $K = 3$*



**RGB**



**CMYK**



**HSV**



*Slika 23. Segmentirane slike pomoću kolornih modela i K-means algoritma pri  $K = 5$*

## 4.1.2 Rezultati uspješnosti segmentacije slika sa prilagođenim kontrastom i osvjetljenjem

U nastavku je detaljno prikazana analiza eksperimentalnih rezultata, koji su ostvareni korištenjem  $K$ -means algoritma u kombinaciji sa kolornim modelima: RGB, CMYK i HSV. Efikasnost segmentacije je evaluirana koristeći četiri različite metrike: Dice koeficijent, F-mjera, Jaccard-ov indeks i Varijacija informacija. Vrijednosti ovih metrika su dobijene usrednjavanjem rezultata po *dataset-u*. Eksperimentalni rezultati realizovani su na odabranom uzorku (*dataset-u*) od 2000 kolornih slika. Slike su prilagođene tako što je faktor kontrasta povećan na 2, a intenzitet osvjetljenja je povećan za 1.5. Dimenzije slika su 400xX piksela, gdje je X varijabilna veličina. Za svaku sliku dostupni su i pripadajući *ground-truth* podaci.

Tabele 3 i 4 i slika 24 detaljno prikazuju rezultate uspješnosti segmentacije slike za dva različita broja klastera:  $K = 3$  i  $K = 5$ . Na slikama 25 i 26 prikazane su segmentirane slike pri  $K=3$  i  $K = 5$ , ilustrujući razlike u segmentaciji u zavisnosti od broja klastera. Slike predstavljaju reprezentativne uzorke iz *dataset-a*.

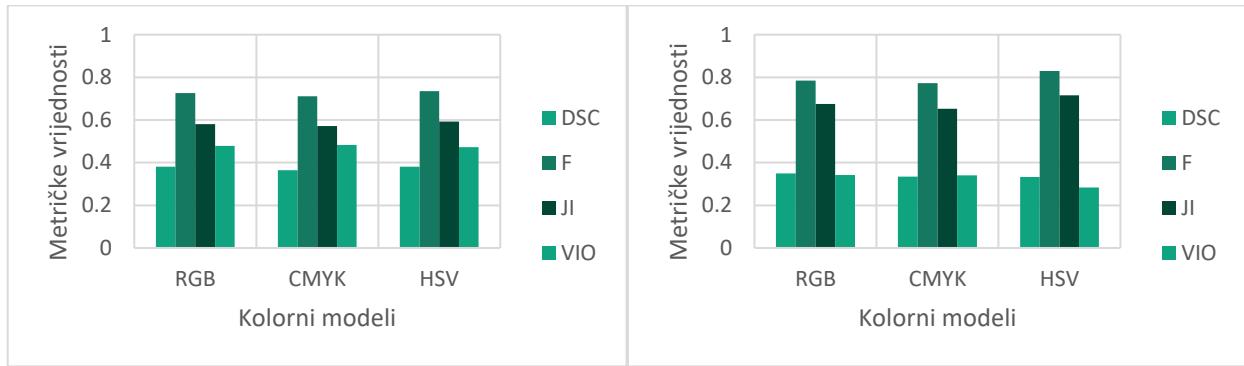
*Tabela 3: Rezultati uspješnosti segmentacije kod K-means algoritma pri  $K = 3$  na dataset-u sa prilagođenim kontrastom i osvjetljenjem*

	DSC	F	JI	VI
<b>RGB</b>	<b>0.3810</b>	0.7264	0.5799	0.4788
<b>CMYK</b>	0.3647	0.7115	0.5718	<b>0.4837</b>
<b>HSV</b>	0.3807	<b>0.7348</b>	<b>0.5932</b>	0.4731

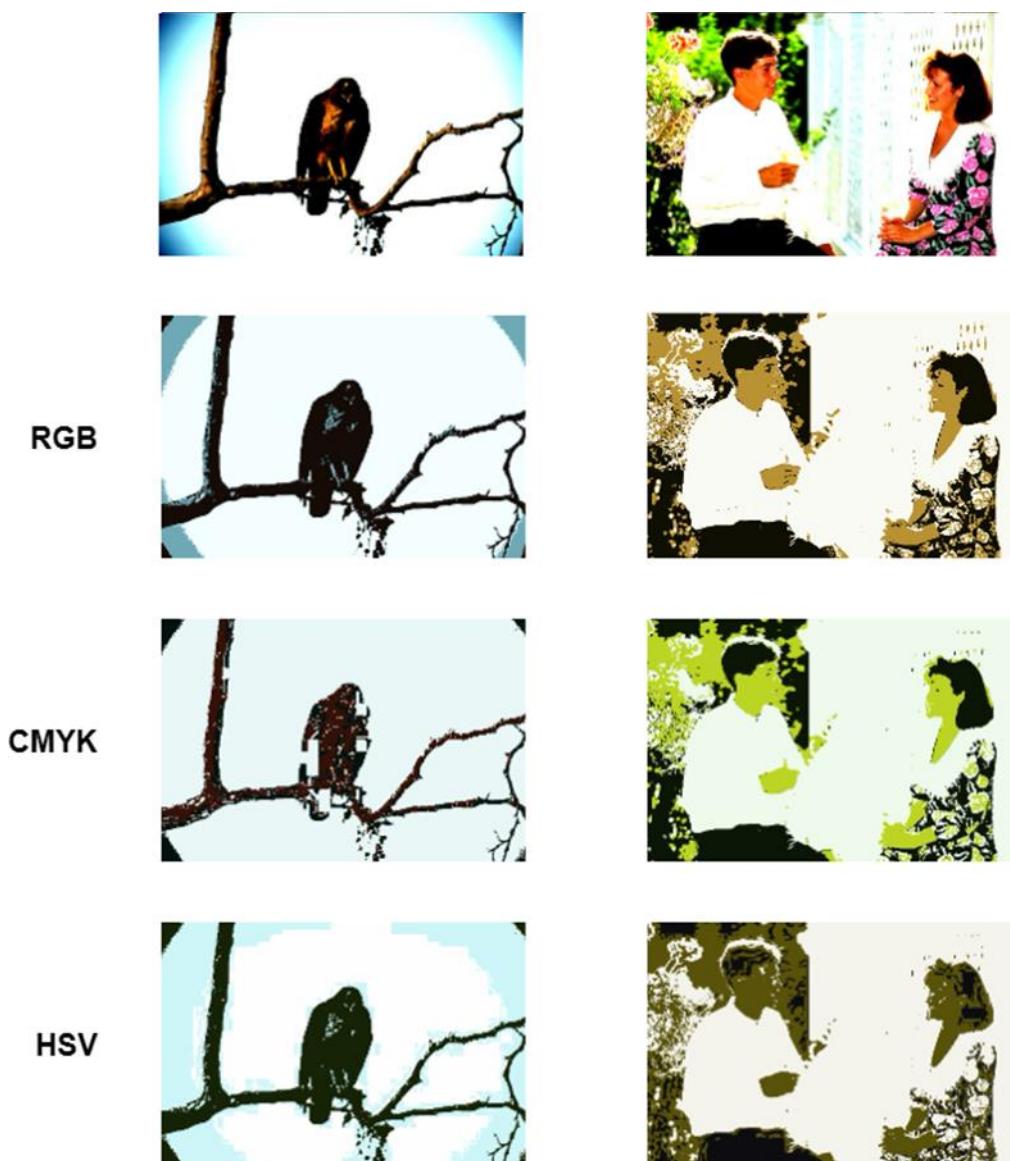
*Tabela 4: Rezultati uspješnosti segmentacije kod K-means algoritma pri  $K = 5$  na dataset-u sa prilagođenim kontrastom i osvjetljenjem*

	DSC	F	JI	VI
<b>RGB</b>	<b>0.3496</b>	0.7849	0.6748	<b>0.3416</b>
<b>CMYK</b>	0.3348	0.7726	0.6531	0.3412
<b>HSV</b>	0.3335	<b>0.8291</b>	<b>0.7162</b>	0.2839

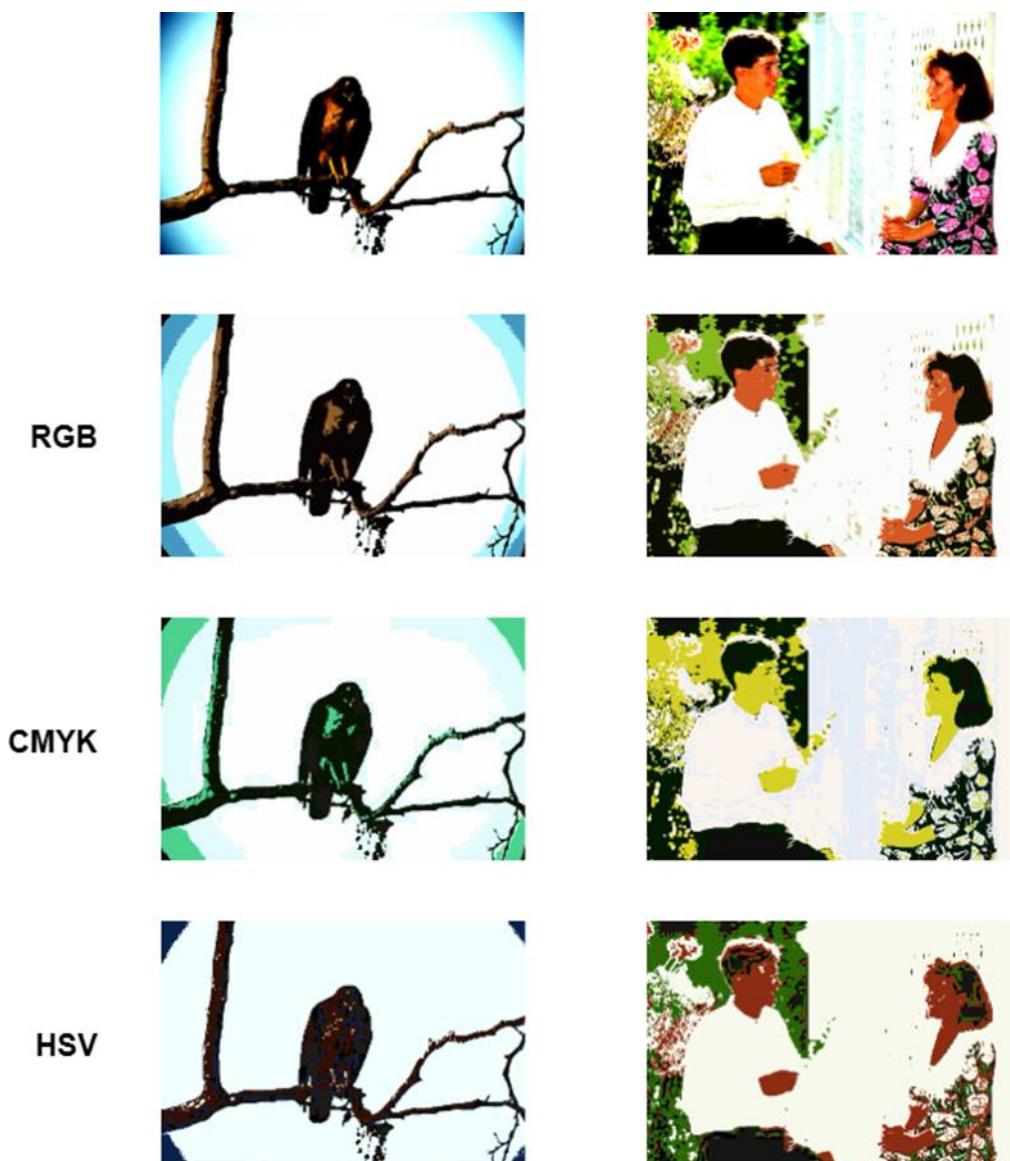
Kod *dataset*-a sa promjenom faktora kontrasta i osvjetljenja ne vidimo neke prevelike razlike u odnosu na *dataset* bez promjena, što pokazuje da promjene kontrasta i osvjetljenja ne utiču puno na rezultate uspješnosti segmentacije korišćenjem *K-means* algoritma i kolornih modela.



*Slika 24. Rezultati uspješnosti segmentacije kod *K-means* algoritma pri  $K = 3$  (lijevo) i pri  $K = 5$  (desno) na dataset-u sa prilagođenim kontrastom i osvjetljenjem*



Slika 25. Segmentirane slike pomoću kolornih modela i K-means algoritma pri  $K=3$  na dataset-u čije su slike prilagođene promjenama kontrasta i osvjetljenja



Slika 26. Segmentirane slike pomoću kolornih modela i K-means algoritma pri  $K=5$  na dataset-u čije su slike prilagođene promjenama kontrasta i osvjetljenja

#### 4.1.3 Rezultati uspješnosti segmentacije slika upotrebom datasetova sa različitim objektima

U nastavku je detaljno prikazana analiza eksperimentalnih rezultata, koji su ostvareni korištenjem  $K$ -means algoritma u kombinaciji sa kolornim modelima: RGB, CMYK i HSV. Efikasnost segmentacije je evaluirana koristeći četiri različite metrike: Dice koeficijent, F-mjera, Jaccard-ov indeks i Varijacija informacija. Vrijednosti ovih metrika su dobijene usrednjavanjem rezultata po *dataset-u*. Eksperimentalni rezultati realizovani su na odabranom uzorku koji obuhvata dva *dataset-a*, svaki sa po 640 slika. Prvi *dataset* obuhvata slike unutrašnjosti kuće, dok drugi *dataset* sadrži slike prirodnih okruženja. Ova podjela je napravljena s ciljem testiranja kako algoritmi izvode segmentaciju fokusiranu na pojedinačne objekte. Dimenzije slika su 400xX piksela, gdje je X varijabilna veličina. Za svaku sliku dostupni su i pripadajući *ground-truth* podaci.

Efikasnost segmentacije pomoću  $K$ -means algoritma ilustruju tabela 7 (za  $K=3$ ) i tabela 8 (za  $K=5$ ), te prateća slika 27, koje se odnose na segmentaciju unutrašnjeg okruženja. S druge strane, tabele 9 i 10 te slika 28 prikazuju rezultate segmentacije za prirodna okruženja, pružajući uvid u različite performanse algoritma. Na slici 29 prikazan je primjer segmentiranih slika unutrašnjih i prirodnih okruženja gdje je  $K = 3$ , dok je na slici 30 prikazan primjer segmentiranih slika unutrašnjih i prirodnih okruženja gdje je  $K = 5$ .

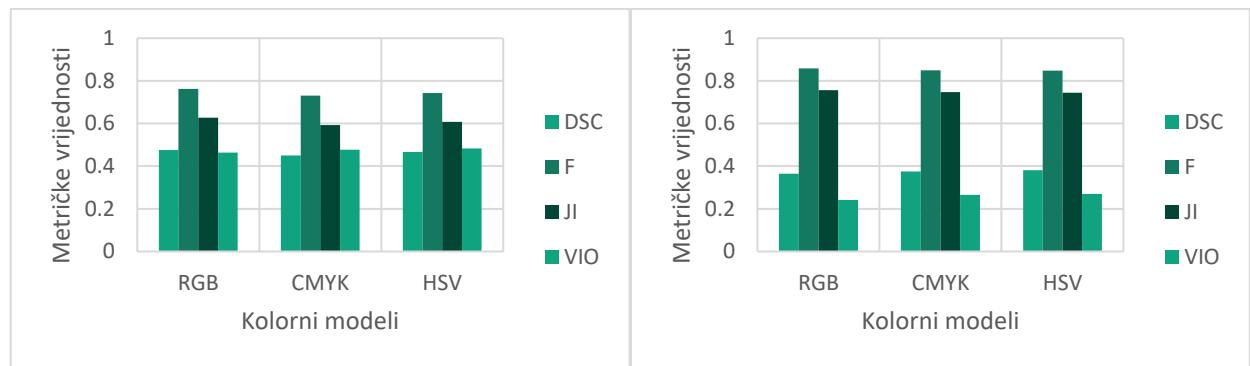
Tabela 5: Rezultati uspješnosti segmentacije kod  $K$ -means algoritma pri  $K = 3$  – unutrašnja okruženja

	DSC	F	JI	VI
<b>RGB</b>	<b>0.4763</b>	<b>0.7628</b>	<b>0.6269</b>	0.4634
<b>CMYK</b>	0.4500	0.7303	0.5923	0.4772
<b>HSV</b>	0.4661	0.7427	0.6079	<b>0.4836</b>

Tabela 6: Rezultati uspješnosti segmentacije kod K-means algoritma pri  $K = 5$  – unutrašnja okruženja

	DSC	F	JI	VI
RGB	0.3643	<b>0.8580</b>	<b>0.7569</b>	0.2423
CMYK	0.3752	0.8491	0.7470	0.2652
HSV	<b>0.3809</b>	0.8475	0.7443	<b>0.2709</b>

Kod  $K = 3$  F-mjera ima najbolji rezultat za sva tri kolorna modela, dok ostale metrike imaju srednju vrijednost. Kod  $K = 5$  uočavaju se poboljšanja kod F-Mjere i Jaccard indeksa, dok je veći pad kod DSC i VI.



Slika 27. Rezultati uspješnosti segmentacije kod K-means algoritma pri  $K = 3$  (lijevo) i pri  $K = 5$  (desno) – unutrašnja okruženja

Tabela 7: Rezultati uspješnosti segmentacije kod K-means algoritma pri  $K = 3$  – prirodna okruženja

	DSC	F	JI	VI
RGB	0.1431	<b>0.7580</b>	0.6224	0.7940
CMYK	<b>0.1560</b>	0.7535	0.6195	<b>0.7997</b>
HSV	0.1500	0.7579	<b>0.6230</b>	0.7918

Tabela 8: Rezultati uspješnosti segmentacije kod K-means algoritma pri  $K = 5$  – prirodna okruženja

	DSC	F	JI	VI
RGB	0.0990	0.8500	0.7456	0.4756
CMYK	0.1051	0.8484	0.7470	<b>0.4924</b>
HSV	<b>0.1056</b>	<b>0.8518</b>	<b>0.7500</b>	0.4830

Za vrijednost  $K = 3$ , primjećuje se da sva tri kolorna modela imaju slabe performanse pri segmentaciji slike kada se koristi DSC metrika za izračunavanje uspješnosti segmentacije, dok ostale metrike pokazuju dobre rezultate. Ovo navodi na pretpostavku da DSC metrika možda nije najprikladnija za ovu vrstu analize.

Upoređujući performanse algoritma između unutrašnjih i prirodnih okruženja, možemo vidjeti da HSV pruža konzistentnije i bolje rezultate za F-mjeru u prirodnim okruženjima, dok je RGB bio bolji u unutrašnjem prostoru prema DSC i JI metrikama. CMYK pokazuje slične ili malo bolje performanse u prirodnim okruženjima u poređenju sa unutrašnjim prostorima, posebno kada je u pitanju VI metrika. Ukupno gledajući, kolorni modeli koji koriste  $K$ -means algoritam pokazuju prednost u segmentaciji slika prirodnih okruženja.



Slika 28. Rezultati uspješnosti segmentacije kod K-means algoritma pri  $K = 3$  (lijevo) i pri  $K = 5$  (desno) – prirodna okruženja



**RGB**



**CMYK**



**HSV**



*Slika 29. Segmentirane slike pomoću kolornih modela i K-means algoritma pri K=3 – prirodna i unutrašnja okruženja.*



**RGB**



**CMYK**



**HSV**



*Slika 30. Segmentirane slike pomoću kolornih modela i K-means algoritma pri K=5 – prirodna i unutrašnja okruženja.*

## 4.2 Otsuov metod

### 4.2.1 Rezultati uspješnosti segmentacije

U nastavku je detaljno prikazana analiza eksperimentalnih rezultata, koji su ostvareni korištenjem Otsuovog metoda u kombinaciji sa kolornim modelima: RGB, CMYK i HSV. Efikasnost segmentacije je evaluirana koristeći četiri različite metrike: Dice koeficijent, F-mjera, Jaccard-ov indeks i Varijacija informacija. Vrijednosti ovih metrika su dobijene usrednjavanjem rezultata po *dataset-u*. Eksperimentalni rezultati realizovani su na odabranom uzorku (*dataset-u*) od 2000 kolornih slika. Slike su obrađene bez promjena kontrasta i osvjetljenja, čime su zadržane u njihovom originalnom vizuelnom stanju. Dimenzije slika su 400xX piksela, gdje je X varijabilna veličina. Za svaku sliku dostupni su i pripadajući *ground-truth* podaci.

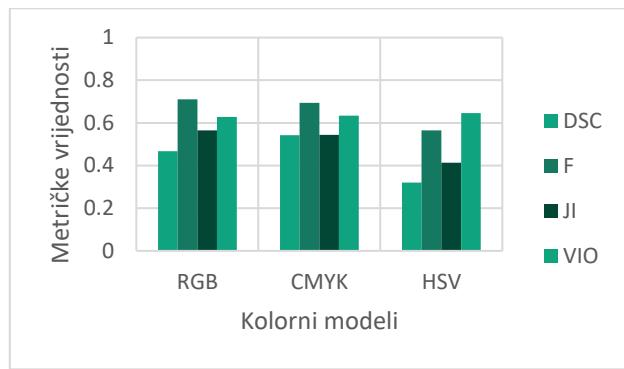
Tabela 5 i slika 31 detaljno prikazuju rezultate uspješnosti segmentacije slike koristeći kolorne modele i Otsuov metod. Na slici 32 prikazane su segmentirane slike pri upotrebi Otsuovog metoda. Slike su izabrane nasumično iz *dataset-a*.

*Tabela 9: Rezultati uspješnosti segmentacije kod Otsuovog metoda na dataset-u originalnih slika*

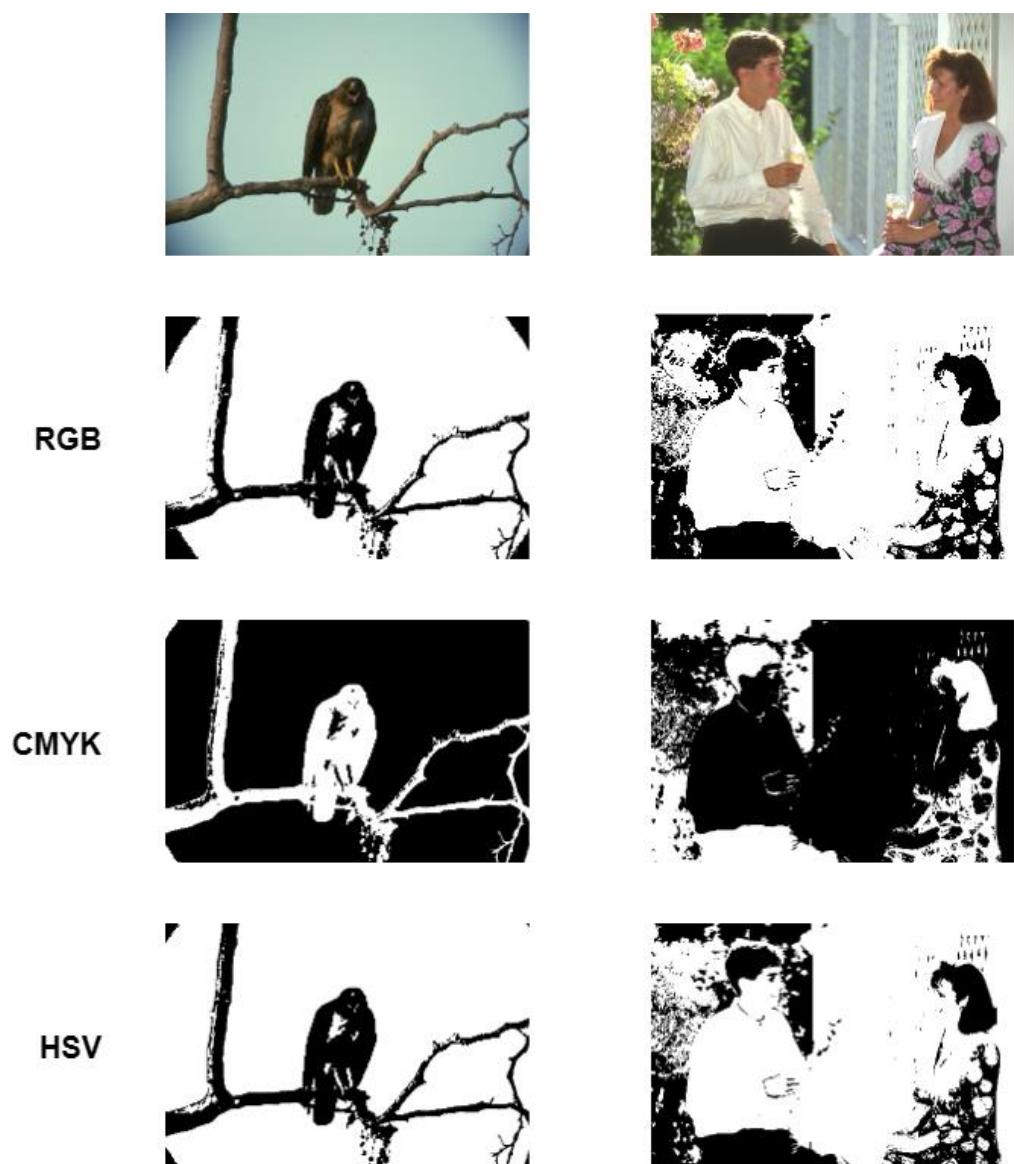
	DSC	F	JI	VI
<b>RGB</b>	0.4671	<b>0.7107</b>	<b>0.5651</b>	0.6278
<b>CMYK</b>	<b>0.5431</b>	0.6944	0.5445	0.6344
<b>HSV</b>	0.3209	0.5645	0.4134	<b>0.6464</b>

Kod Otsuove metode, CMYK model se ističe po VI, što ukazuje na to da je dobro strukturirao informacije prilikom segmentacije, dok RGB model pokazuje solidne performanse preko svih metrika. HSV model ima neke slabosti u preciznosti i JI, ali ima visok VI.

Kada se uporedi sa *K-means* rezultatima, primjećuje se da Otsuov metod generalno čini da pruža bolju informacijsku konzistentnost (VI) za sve modele, što može biti posljedica njegove automatizovane prirode u određivanju optimalnog praga za segmentaciju, umjesto subjektivnog odabira broja klastera kao u *K-means* metodi. Ovo može ukazivati na to da je Otsuov metod bolji izbor kada je struktura podataka manje poznata ili kada je teško odlučiti o pravom broju klastera.



Slika 31. Rezultati uspješnosti segmentacije kod Otsuovog metoda na dataset-u originalnih slika



Slika 32. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda na dataset-u originalnih slika.

## 4.2.2 Rezultati uspješnosti segmentacije slika sa prilagođenim kontrastom i osvjetljenjem

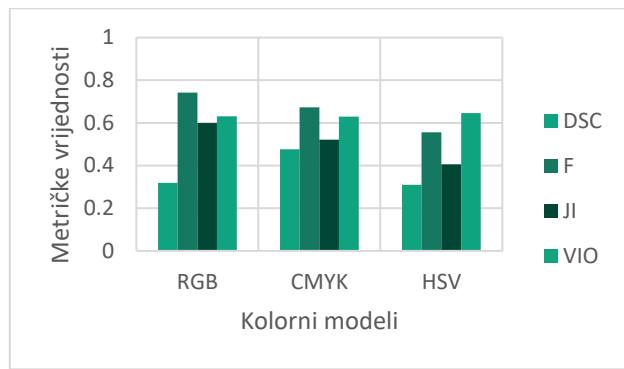
U nastavku je detaljno prikazana analiza eksperimentalnih rezultata, koji su ostvareni korištenjem Otsuovog metoda u kombinaciji sa kolornim modelima: RGB, CMYK i HSV. Efikasnost segmentacije je evaluirana koristeći četiri različite metrike: Dice koeficijent, F-mjera, Jaccard-ov indeks i Varijacija informacija. Vrijednosti ovih metrika su dobijene usrednjavanjem rezultata po *dataset-u*. Eksperimentalni rezultati realizovani su na odabranom uzorku (*dataset-u*) od 2000 kolornih slika. Slike su prilagodene tako što je faktor kontrasta povećan na 2, a intenzitet osvjetljenja je povećan za 1.5. Dimenzije slika su 400xX piksela, gdje je X varijabilna veličina. Za svaku sliku dostupni su i pripadajući *ground-truth* podaci.

Tabele 6 i slika 33 detaljno prikazuju rezultate uspješnosti segmentacije slike koristeći kolorne modele i Otsuov metod. Na slici 34 prikazane su segmentirane slike pri upotrebi Otsuovog metoda. Slike su izabrane nasumično iz *dataset-a* gdje je promijenjen faktor za kontrast i osvjetljenje.

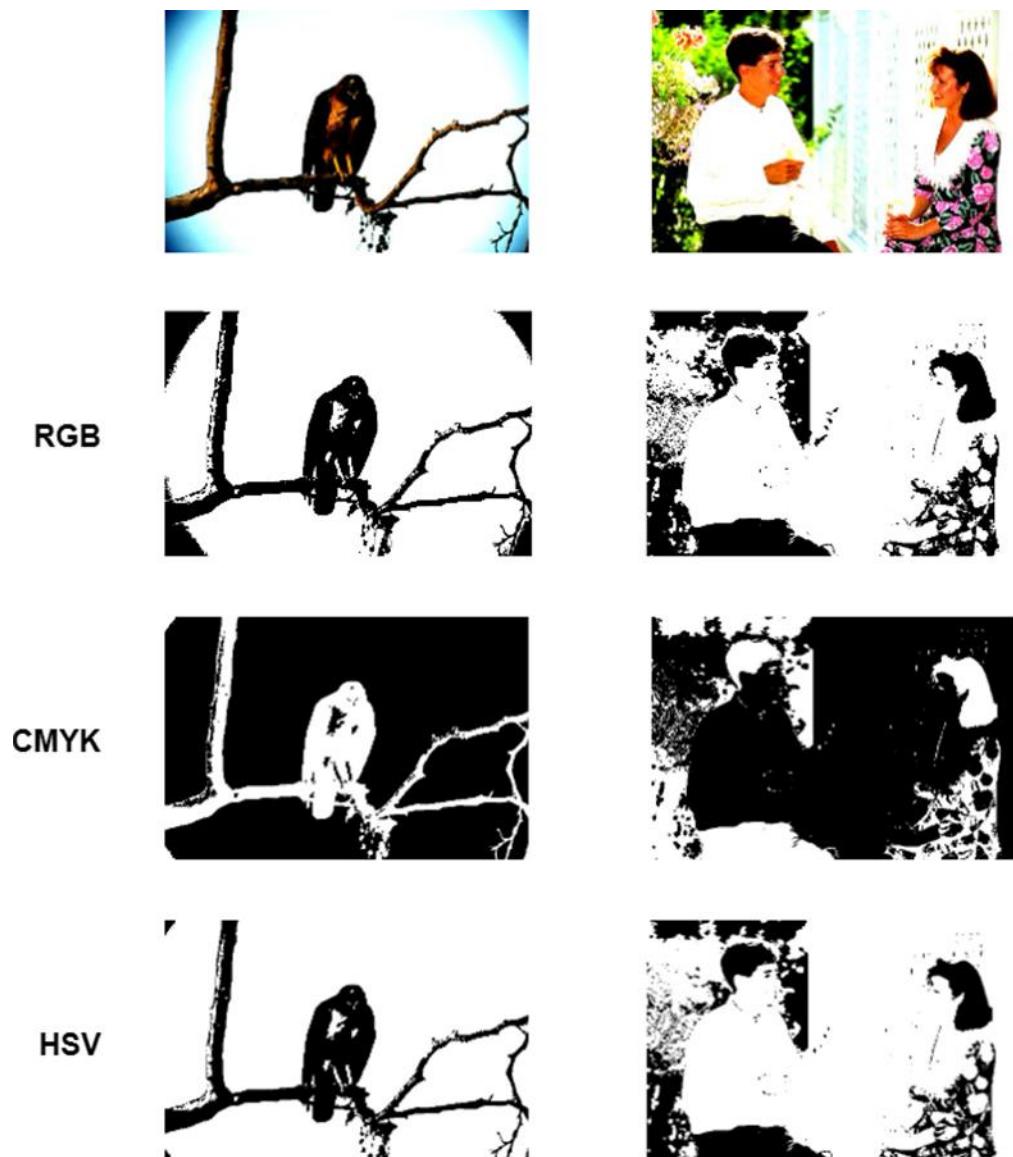
*Tabela 10: Rezultati uspješnosti segmentacije kod Otsuovog metoda na dataset-u sa prilagođenim kontrastom i osvjetljenjem*

	DSC	F	JI	VI
RGB	0.3189	<b>0.7426</b>	<b>0.6001</b>	0.6306
CMYK	<b>0.4761</b>	0.6730	0.5214	0.6298
HSV	0.3091	0.5564	0.4056	<b>0.6458</b>

Kod *dataset-a* sa promjenom faktora kontrasta i osvjetljenja ne vidimo neke prevelike razlike u odnosu na *dataset* bez promjena, što pokazuje da promjene kontrasta i osvjetljenja ne utiču puno na rezultate uspješnosti segmentacije korišćenjem Otsuovog metoda i kolornih modela.



*Slika 33. Rezultati uspješnosti segmentacije kod Otsuovog metoda na dataset-u sa prilagođenim kontrastom i osvjetljenjem*



*Slika 34. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda na dataset-u čije su slike prilagođene promjenama kontrasta i osvjetljenja.*

## 4.2.3 Rezultati uspješnosti segmentacije slika upotrebom datasetova sa različitim objektima

U nastavku je detaljno prikazana analiza eksperimentalnih rezultata, koji su ostvareni korištenjem Otsuovog metoda u kombinaciji sa kolornim modelima: RGB, CMYK i HSV. Efikasnost segmentacije je evaluirana koristeći četiri različite metrike: Dice koeficijent, F-mjera, Jaccard-ov indeks i Varijacija informacija. Vrijednosti ovih metrika su dobijene usrednjavanjem rezultata po *dataset-u*. Eksperimentalni rezultati realizovani su na odabranom uzorku koji obuhvata dva *dataset-a*, svaki sa po 640 slika. Prvi *dataset* obuhvata slike unutrašnjosti kuće, dok drugi *dataset* sadrži slike prirodnih okruženja. Ova podjela je napravljena s ciljem testiranja kako algoritmi izvode segmentaciju fokusiranu na pojedinačne objekte. Dimenzije slika su 400xX piksela, gdje je X varijabilna veličina. Za svaku sliku dostupni su i pripadajući *ground-truth* podaci.

Efikasnost segmentacije pomoću Otsuovog metoda ilustruju tabela 11, te prateći slika 35 (lijevo), koji se odnose na segmentaciju unutrašnjeg okruženja. S druge strane, tabela 12 i slika 35 (desno) prikazuju rezultate segmentacije za prirodna okruženja, pružajući uvid u različite performanse algoritma. Na slici 36 prikazani su primjeri segmentiranih slika. Slike su izabrane nasumično iz *dataset-a*.

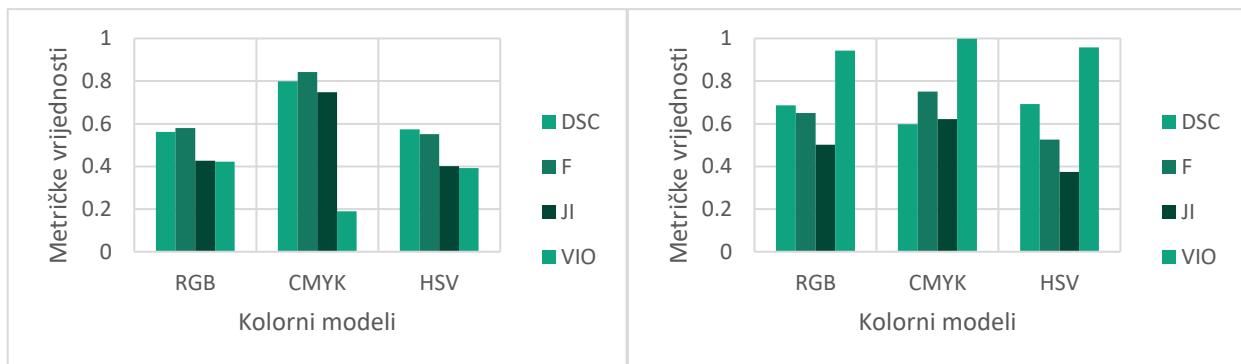
Tabela 11: Rezultati uspješnosti segmentacije kod Otsuovog metoda – unutrašnja okruženja

	DSC	F	JI	VI
RGB	0.5612	0.5805	0.4267	<b>0.4221</b>
CMYK	<b>0.7983</b>	<b>0.8429</b>	<b>0.7484</b>	0.1895
HSV	0.5740	0.5520	0.4010	0.3926

Tabela 12: Rezultati uspješnosti segmentacije kod Otsuovog metoda – prirodna okruženja

	DSC	F	JI	VI
RGB	0.6872	0.6497	0.5013	0.9427
CMYK	0.5972	<b>0.7507</b>	<b>0.6225</b>	<b>0.9990</b>
HSV	<b>0.6931</b>	0.5253	0.3746	0.9588

Analizirajući rezultate vidimo da se Otsuov metod bolje snašao pri segmentaciji slika iz prirodnog okruženja, gdje je metrike kod VI blizu perfektnе uspješnosti segmentacije, dok su i ostale metrike na zavidnom nivou.



*Slika 35. Rezultati uspješnosti segmentacije kod Otsuovog metoda – unutrašnja okruženja (lijevo) i prirodna okruženja (desno)*



**RGB**



**CMYK**



**HSV**



*Slika 36. Segmentirane slike pomoću kolornih modela i Otsuovog metoda – prirodna i unutrašnja okruženja.*

## 4.3 Otsuov metod sa višestrukim pragovima

### 4.3.1 Rezultati uspješnosti segmentacije

U nastavku je detaljno prikazana analiza eksperimentalnih rezultata, koji su ostvareni korištenjem Otsuovog metoda sa višestrukim pragovima u kombinaciji sa kolornim modelima: RGB, CMYK i HSV. Efikasnost segmentacije je evaluirana koristeći četiri različite metrike: Dice koeficijent, F-mjera, Jaccard-ov indeks i Varijacija informacija. Vrijednosti ovih metrika su dobijene usrednjavanjem rezultata po *dataset-u*. Eksperimentalni rezultati realizovani su na odabranom uzorku (*dataset-u*) od 2000 kolornih slika. Slike su obrađene bez promjena kontrasta i osvjetljenja, čime su zadržane u njihovom originalnom vizuelnom stanju. Dimenzije slika su 400xX piksela, gdje je X varijabilna veličina. Za svaku sliku dostupni su i pripadajući *ground-truth* podaci.

Tabele 13 i 14 i slika 37 detaljno prikazuju rezultate uspješnosti segmentacije slike za dva različita broja klasa: 3 i 5. Na slikama 38 i 39 prikazane su segmentirane slike sa 3 klase i sa 5 klase, ilustrujući razlike u segmentaciji u zavisnosti od broja klasa. Slike predstavljaju reprezentativne uzorke iz *dataset-a*.

*Tabela 13. Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (3 klase) na dataset-u originalnih slika*

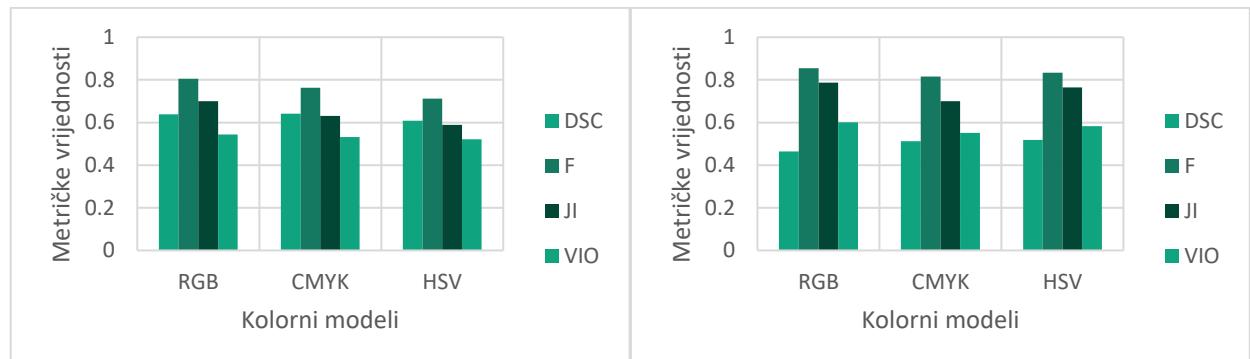
	DSC	F	JI	VI
RGB	0.6389	<b>0.8045</b>	<b>0.7003</b>	0.5442
CMYK	<b>0.6416</b>	0.7626	0.6301	0.5318
HSV	0.6077	0.7118	0.5890	<b>0.5215</b>

Tabela 14: Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (5 klase) na dataset-u originalnih slika.

	DSC	F	JI	VI
RGB	0.4649	<b>0.8546</b>	<b>0.7874</b>	<b>0.6013</b>
CMYK	0.5123	0.8148	0.7000	0.5509
HSV	<b>0.5186</b>	0.8339	0.7642	0.5832

U tabeli 13, gdje je primijenjen Otsuov metod sa višestrukim pragovima sa 3 klase na RGB slikama, DSC pokazuje solidan rezultat, što implicira visoku preciznost segmentacije. F-mjera je prilično visoka, što ukazuje na dobar balans između preciznosti i odziva. JI ukazuje na solidnu uspješnost, odnosno pristojno preklapanje između segmentiranih i stvarnih regija. VI je srednje vrijednosti, što znači da postoji prostor za poboljšanje u smislu informacijske konzistentnosti. Kod CMYK-a i HSV-a, vidimo slične vrijednosti za VI, dok su ostale metrike u blagom padu.

U tabeli 14, gdje je primijenjen Otsuov metod sa višestrukim pragovima sa 5 klase, F-mjera je ostala na sličnom nivou kod sva tri kolorna modela, VI je nešto slabiji, JI pokazuje pad kod HSV i CMYK kolornog modela, dok DSC pokazuje bolje rezultate kod sva tri kolorna modela.



Slika 37. Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višeststrukim pragovima - 3 klase (lijevo), 5 klase (desno) na dataset-u originalnih slika



**RGB**



**CMYK**



**HSV**



*Slika 38. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda sa višestrukim pragovima (3 klase) na dataset-u originalnih slika.*



**RGB**



**CMYK**



**HSV**



*Slika 39. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda sa višestrukim pragovima (5 klasa) na dataset-u originalnih slika.*

## 4.3.2 Rezultati uspješnosti segmentacije slika sa prilagođenim kontrastom i osvjetljenjem

U nastavku je detaljno prikazana analiza eksperimentalnih rezultata, koji su ostvareni korištenjem Otsuovog metoda sa višestrukim pragovima u kombinaciji sa kolornim modelima: RGB, CMYK i HSV. Efikasnost segmentacije je evaluirana koristeći četiri različite metrike: Dice koeficijent, F-mjera, Jaccard-ov indeks i Varijacija informacija. Vrijednosti ovih metrika su dobijene usrednjavanjem rezultata po *dataset-u*. Eksperimentalni rezultati realizovani su na odabranom uzorku (*dataset-u*) od 2000 kolornih slika. Slike su prilagođene tako što je faktor kontrasta povećan na 2, a intenzitet osvjetljenja je povećan za 1.5. Dimenzije slika su 400xX piksela, gdje je X varijabilna veličina. Za svaku sliku dostupni su i pripadajući *ground-truth* podaci.

Tabele 15 i 16 i slika 40 detaljno prikazuju rezultate uspješnosti segmentacije slike za dva različita broja klase: 3 i 5. Na slikama 41 i 42 prikazane su segmentirane slike sa 3 klase i sa 5 klase, ilustrujući razlike u segmentaciji u zavisnosti od broja klasa. Slike predstavljaju reprezentativne uzorke iz *dataset-a*.

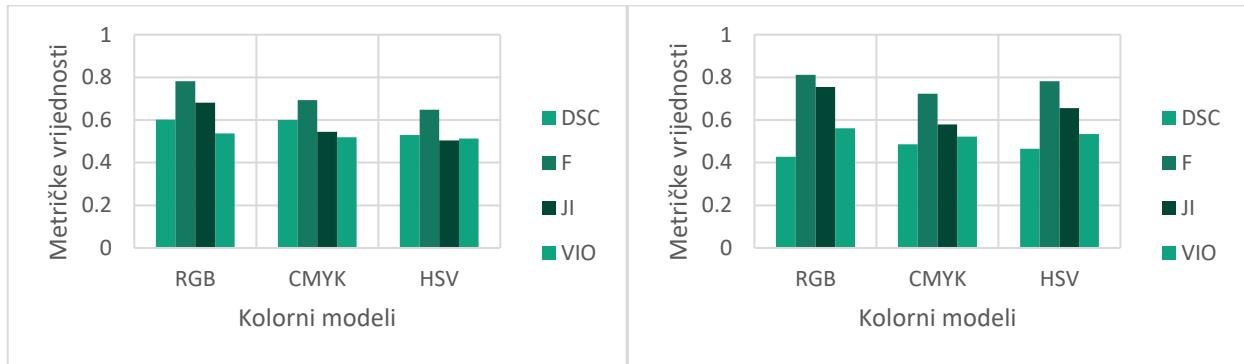
*Tabela 15: Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (3 klase) na dataset-u čije su slike prilagođene promjenama kontrasta i osvjetljenja*

	DSC	F	JI	VI
<b>RGB</b>	<b>0.6009</b>	<b>0.7817</b>	<b>0.6808</b>	0.5375
<b>CMYK</b>	0.6005	0.6939	0.5449	0.5189
<b>HSV</b>	0.5298	0.6482	0.5046	<b>0.5129</b>

*Tabela 16: Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (5 klase) na dataset-u čije su slike prilagođene promjenama kontrasta i osvjetljenja*

	DSC	F	JI	VI
<b>RGB</b>	0.4271	<b>0.8111</b>	<b>0.7550</b>	0.5615
<b>CMYK</b>	<b>0.4856</b>	0.7231	0.5796	<b>0.5223</b>
<b>HSV</b>	0.4656	0.7823	0.6556	0.5342

Kod *dataset*-a sa promjenom faktora kontrasta i osvjetljenja ne vidimo neke prevelike razlike u odnosu na *dataset* bez promjena, što pokazuje da promjene kontrasta i osvjetljenja ne utiču puno na rezultate uspješnosti segmentacije korišćenjem Otsuovog metoda sa višestrukim pragovima i kolornih modela.



*Slika 40. Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima - 3 klase (lijevo), 5 klase (desno) na dataset-u čije su slike prilagođene promjenama kontrasta i osvjetljenja*



**RGB**



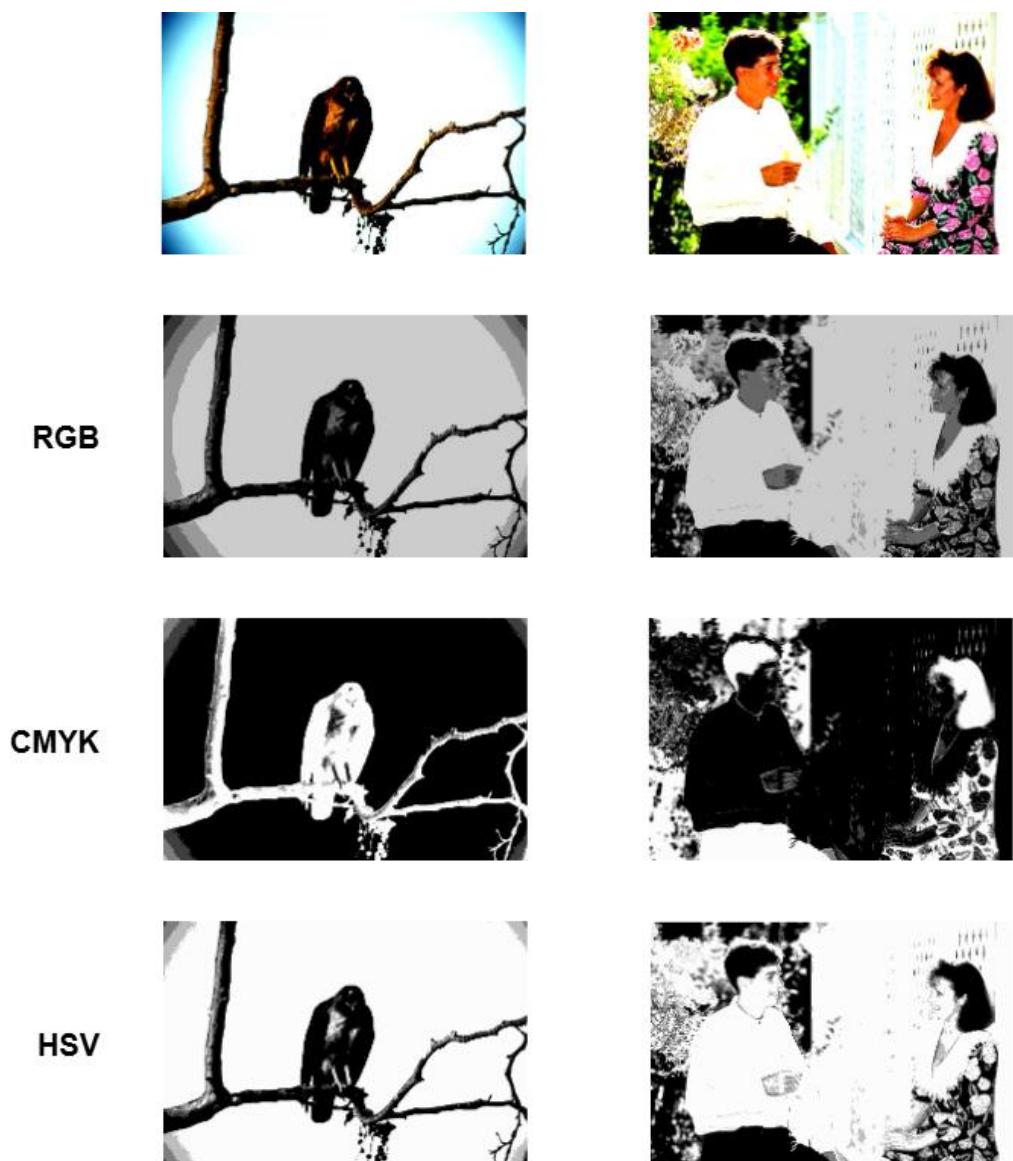
**CMYK**



**HSV**



*Slika 41. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda sa višestrukim pragovima (3 klase) na dataset-u čije su slike prilagođene promjenama kontrasta i osvjetljenja.*



Slika 42. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda sa višestrukim pragovima (5 klase) na dataset-u čije su slike prilagođene promjenama kontrasta i osvjetljenja.

### 4.3.3 Rezultati uspješnosti segmentacije slika upotrebom datasetova sa različitim objektima

U nastavku je detaljno prikazana analiza eksperimentalnih rezultata, koji su ostvareni korištenjem Otsuovog metoda sa višestrukim pragovima u kombinaciji sa kolornim modelima: RGB, CMYK i HSV. Efikasnost segmentacije je evaluirana koristeći četiri različite metrike: Dice koeficijent, F-mjera, Jaccard-ov indeks i Varijacija informacija. Vrijednosti ovih metrika su dobijene usrednjavanjem rezultata po *dataset-u*. Eksperimentalni rezultati realizovani su na odabranom uzorku koji obuhvata dva *dataset-a*, svaki sa po 640 slika. Prvi *dataset* obuhvata slike unutrašnjosti kuće, dok drugi *dataset* sadrži slike prirodnih okruženja. Ova podjela je napravljena s ciljem testiranja kako algoritmi izvode segmentaciju fokusiranu na pojedinačne objekte. Dimenzije slika su 400xX piksela, gdje je X varijabilna veličina. Za svaku sliku dostupni su i pripadajući *ground-truth* podaci.

Efikasnost segmentacije pomoću Otsuovog metoda sa višestrukim pragovima ilustruju tabela 17 i tabela 18, te prateća slika 43, koje se odnose na segmentaciju unutrašnjeg okruženja. S druge strane, tabele 19 i 20, te prateća slika 44 prikazuju rezultate segmentacije za prirodna okruženja, pružajući uvid u različite performanse algoritma. Na slici 45 prikazan je primjer segmentiranih slika unutrašnjih i prirodnih okruženja gdje je slika segmentirana u 3 klase, dok je na slici 46 prikazan primjer segmentiranih slika unutrašnjih i prirodnih okruženja gdje je slika segmentirana u 5 klasa.

Tabela 17: Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (3 klase) – unutrašnja okruženja

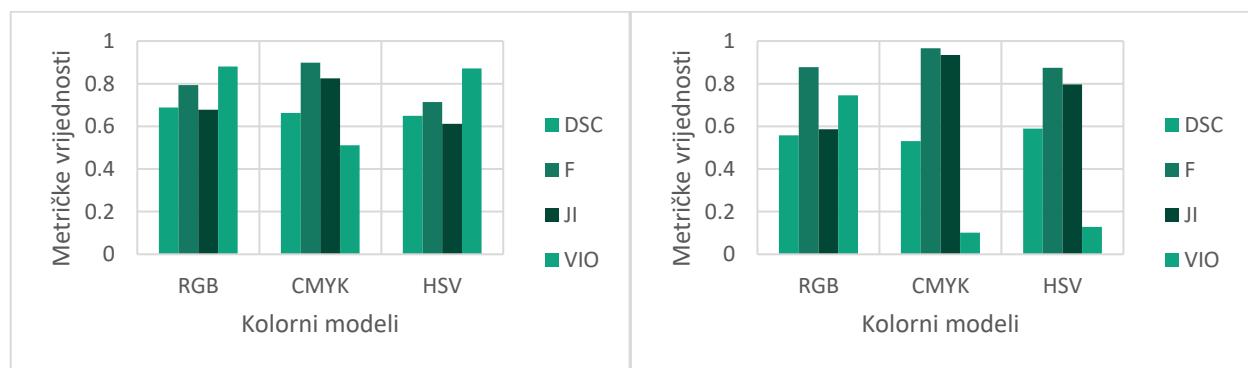
	DSC	F	JI	VI
<b>RGB</b>	<b>0.6882</b>	0.7929	0.6773	<b>0.8800</b>
<b>CMYK</b>	0.6635	<b>0.8987</b>	<b>0.8256</b>	0.5119
<b>HSV</b>	0.6491	0.7134	0.6116	0.8708

Tabela 18: Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (5 klase) – unutrašnja okruženja

	DSC	F	JI	VI
<b>RGB</b>	0.5576	0.8769	0.5866	<b>0.7448</b>
<b>CMYK</b>	0.5301	<b>0.9654</b>	<b>0.9350</b>	0.1013
<b>HSV</b>	<b>0.5892</b>	0.8746	0.7972	0.1292

U tabeli 17, gdje je primijenjen Otsuov metod sa višestrukim pragovima sa 3 klase na unutrašnje okruženje, DSC pokazuje najbolje rezultate kod RGB modela, što ukazuje na visoku preciznost segmentacije. F-mjera je najviša kod CMYK modela, što znači da ovaj model postiže najbolji balans između preciznosti i odziva. JI takođe ima najbolji rezultat kod CMYK modela, što ukazuje na najveće preklapanje između segmentiranih i stvarnih regija. VI je najbliži 1 kod RGB modela, što znači da je informacijska konzistentnost najbolja kod ovog modela.

U tabeli 18, gdje je primijenjen Otsuov metod sa višestrukim pragovima sa 5 klasa na unutrašnje okruženje, F-mjera ostaje na visokom nivou kod sva tri kolorna modela, sa CMYK modelom koji ima najbolji rezultat. VI je nešto slabiji u poređenju sa tabelom 17, ali RGB model i dalje postiže relativno dobar rezultat. JI ima pad kod HSV i CMYK modela, dok DSC pokazuje poboljšane rezultate kod HSV modela, što ukazuje na poboljšanu preciznost segmentacije kod ovog modela.



Slika 43. Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima - 3 klase (lijevo), 5 klase (desno) – unutrašnja okruženja

Tabela 19: Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (3 klase) – prirodna okruženja

	DSC	F	JI	VI
RGB	0.6273	0.7163	0.5714	0.3001
CMYK	<b>0.6600</b>	<b>0.9433</b>	<b>0.8964</b>	<b>0.5030</b>
HSV	0.6439	0.7317	0.5938	0.3453

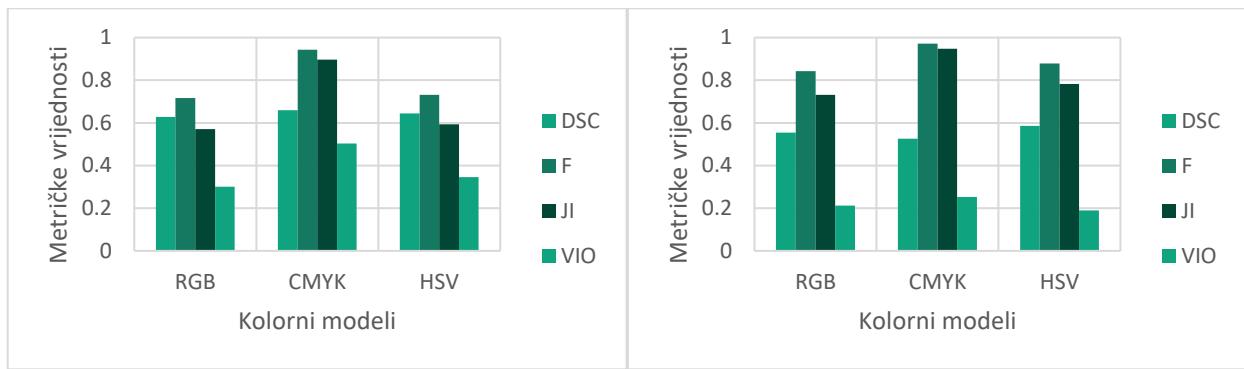
Tabela 20: Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima (5 klase) – prirodna okruženja

	DSC	F	JI	VI
RGB	0.5539	0.8428	0.7318	0.2121
CMYK	0.5266	<b>0.9723</b>	<b>0.9479</b>	<b>0.2527</b>
HSV	<b>0.5866</b>	0.8783	0.7820	0.1891

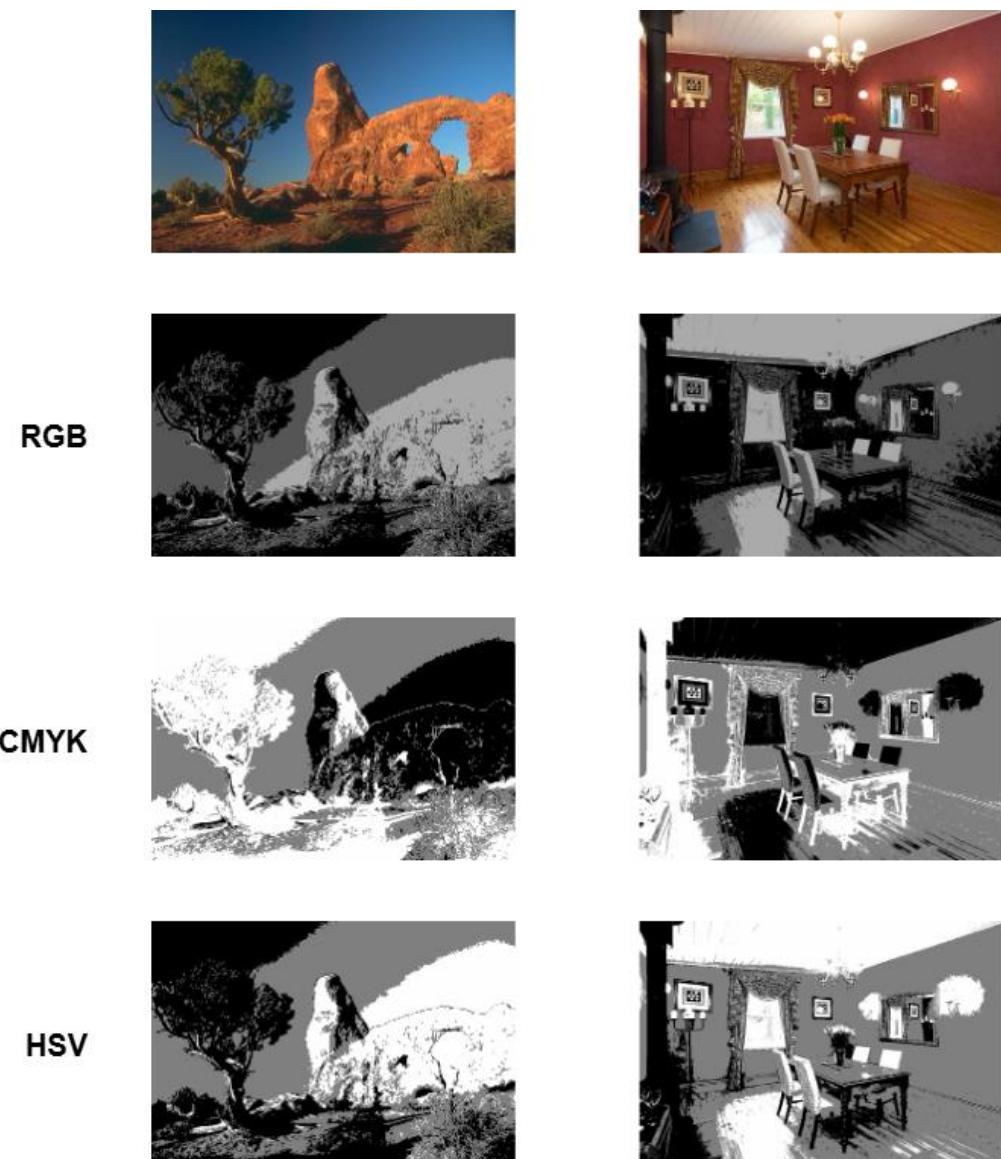
U tabeli 19, primjenom Otsuovog metoda sa višestrukim pragovima sa 3 klase na prirodna okruženja, CMYK model ima najbolje rezultate za DSC, F-mjeru, JI i VI.

U tabeli 20, sa višestrukim pragovima sa 5 klase, CMYK model i dalje pokazuje najbolje rezultate za F-mjeru i JI, dok RGB i HSV modeli pokazuju nešto slabije performanse. DSC rezultati su poboljšani kod HSV modela.

Upoređujući performanse Otsuovog metoda između unutrašnjih i prirodnih okruženja, možemo vidjeti različite učinke kolornog modela. U unutrašnjim okruženjima, RGB model pokazuje najbolje rezultate za DSC i JI metrike, dok F-mjera ostaje visoka kod CMYK modela. HSV model ima poboljšane rezultate za DSC kada se koristi sa 5 klase. U prirodnim okruženjima, CMYK model dominira sa najboljim rezultatima za DSC, F-mjeru, JI i VI. RGB model pokazuje bolje rezultate u unutrašnjim okruženjima prema DSC i JI metrikama, dok je u prirodnim okruženjima nešto slabiji. HSV model pruža konzistentnije performanse, sa poboljšanim DSC rezultatima u oba okruženja, ali pokazuje bolje rezultate za F-mjeru u prirodnim okruženjima. Sve u svemu, CMYK model pokazuje blagu prednost u prirodnim okruženjima.



Slika 44. Rezultati uspješnosti segmentacije kod Otsuovog metoda sa višestrukim pragovima - 3 klase (lijevo), 5 klasa (desno) – prirodna okruženja



Slika 45. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda sa višestrukim pragovima (3 klase) – prirodna i unutrašnja okruženja.



**RGB**



**CMYK**



**HSV**



*Slika 46. Segmentirane slike uz pomoć kolornih modela i Otsuovog metoda sa višestrukim pragovima (5 klase) – prirodna i unutrašnja okruženja.*

# Zaključak

Detaljna analiza obuhvatila je složeno područje segmentacije slike gdje su odnosi između kolornih modela i algoritama segmentacije postali očigledniji. Napravljena je složena analiza tri kolorna modela – RGB, HVS i CMYK, zajedno sa tri algoritma segmentacije, tj. K-means, Otsuov metod i Otsuov metod sa višestrukim pragovima. Validacija rezultata je obavljena korišćenjem četiri metrike uspješnosti, odnosno kvaliteta, segmentacije: Dice koeficijent, Jaccard-ov indeks, F-mjera i Validacija informacija.

Rezultati su otkrili zanimljive obrasce. Osim toga, upotreba različitih kolornih modela sa Otsuovom metodom rezultira promjenljivim performansama segmentacije, posebno u kontekstu okruženja slike. Suptilno korišćenje boja u različitim šemama boja dalo je prostora za slike okružene prirodnim scenama, uz suptilnu ravnotežu između boja u različitim kolornim modelima i njihovog uspjeha u segmentaciji. S druge strane, slike unutrašnjih područja bile su bolje usklađene sa K-means algoritmom koji je dokazao njegovu efikasnost u ovim slučajevima.

Kod Otsuovog metoda sa višestrukim pragovima eksperimentisalo se sa različitim brojem klasa kako bi se razumio njihov uticaj na rezultate segmentacije. Rezultati su pokazali da i K-means i Otsuov metod sa višestrukim pragovima daju slične rezultate. Međutim, u poređenju sa standardnim Otsuovim metodom, koji dijeli sliku na dvije klase, Otsuov metod sa višestrukim pragovima pokazao je bolje rezultate. To je očekivano jer segmentacija sa više klasa omogućava detaljniju analizu slike i pruža bolje performanse u različitim scenarijima.

Jedno značajno zapažanje iz rezultata je snažna prilagodljivost sva tri algoritma prema promjenama u osvjetljenju i kontrastu. Robusnost ovih algoritama je očigledna jer se mogu primijeniti u mnogim i teškim okruženjima.

Ipak, zapaženo je da nijedan od algoritama nije imao izuzetne performanse u svim scenarijima. Ovo pokazuje da može da postoji određeni nivo osjetljivosti ovih algoritama na izbor kolornog modela. Uspješna segmentacija takođe zavisi od specifičnosti skupa podataka.

Iako se segmentacija, uključujući semantičku, često izvodi pomoću složenih neuralnih mreža poput konvolucijskih, ne treba zanemariti značaj jednostavnijih. Ovi algoritmi i dalje imaju ključnu ulogu u brojnim primjenama zbog svoje efikasnosti i jednostavnosti u upotrebi.

# Appendix (dodatak)

## Dodatak A: *K*-means algoritam

```
# K-means

from sklearn.cluster import KMeans
from skimage.io import imread
import matplotlib.pyplot as plt

def segment_image_with_kmeans(image_path, n_clusters=3):
    image = imread(image_path)
    rows, cols, _ = image.shape

    image_reshaped = image.reshape(rows * cols, -1)

    kmeans = KMeans(n_clusters=n_clusters, n_init=10,
                     random_state=42)
    kmeans.fit(image_reshaped)

    labels = kmeans.labels_
    segmented_image = labels.reshape(rows, cols)

    return segmented_image

def show_original_and_segmented(original_path, segmented_image):
    original_image = imread(original_path)

    plt.figure(figsize=(12, 6))

    plt.subplot(1, 2, 1)
    plt.imshow(original_image)
    plt.title('Original Image')
    plt.axis('off')

    plt.subplot(1, 2, 2)
    plt.imshow(segmented_image, cmap='viridis')
    plt.title('Segmented Image')
    plt.axis('off')

    plt.show()

image_path = "path.jpg"
segmented_image = segment_image_with_kmeans(image_path,
                                             n_clusters=3)
show_original_and_segmented(image_path, segmented_image)
```

## Dodatak B: Otsuov metod

```
# Otsuov metod

import numpy as np
import cv2
from skimage.io import imread
from skimage.color import rgb2gray
import matplotlib.pyplot as plt

def segment_image_otsu(image):
    gray_image = rgb2gray(image)

    _, segmented_image = cv2.threshold((gray_image * 255).astype(np.uint8), 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)

    return segmented_image

def show_original_and_segmented(original_path, segmented_image):
    original_image = imread(original_path)

    plt.figure(figsize=(12, 6))
    plt.subplot(1, 2, 1)
    plt.imshow(original_image)
    plt.title('Original Image')
    plt.axis('off')

    plt.subplot(1, 2, 2)
    plt.imshow(segmented_image, cmap='gray')
    plt.title('Segmented Image')
    plt.axis('off')

    plt.show()

image_path = "path.jpg"
original_image = imread(image_path)
segmented_image = segment_image_otsu(original_image)
show_original_and_segmented(image_path, segmented_image)
```

## Dodatak C: Otsuov metod sa višestrukim pragovanjem

```
# Otsuov metod sa višestrukim pragovanjem

import numpy as np
from skimage.io import imread
from skimage.color import rgb2gray
from skimage.filters import threshold_multiotsu
import matplotlib.pyplot as plt

def segment_image_multi_otsu(image, classes):
    gray_image = rgb2gray(image)

    thresholds = threshold_multiotsu(gray_image, classes=classes)
    regions = np.digitize(gray_image, bins=thresholds)

    segmented_image = (regions * (255 /
regions.max())).astype(np.uint8)

    return segmented_image

def show_original_and_segmented(original_path, segmented_image):
    original_image = imread(original_path)

    plt.figure(figsize=(12, 6))
    plt.subplot(1, 2, 1)
    plt.imshow(original_image)
    plt.title('Original Image')
    plt.axis('off')

    plt.subplot(1, 2, 2)
    plt.imshow(segmented_image, cmap='gray')
    plt.title('Segmented Image')
    plt.axis('off')

    plt.show()

image_path = "path.jpg"
original_image = imread(image_path)
classes = 3
segmented_image = segment_image_multi_otsu(original_image, classes)
show_original_and_segmented(image_path, segmented_image)
```

## Dodatak D: Metode validacije

```
# Dice Coefficient

import numpy as np

def calculate_dice_coefficient(segmented, reference):
    intersection = np.logical_and(segmented, reference)
    dice_coefficient = 2 * intersection.sum() / (segmented.sum() +
reference.sum())
    return dice_coefficient

# Primjer upotrebe
# Ovdje bi se učitala segmentirana slika i referentna slika
# i zatim pozvala funkcija calculate_dice_coefficient

# dice_score = calculate_dice_coefficient(segmented, reference)
# print(f"Dice Coefficient: {dice_score:.4f}")
```

```
# F-Measure

import numpy as np

def calculate_f_measure(segmented, reference):
    TP = np.logical_and(segmented, reference).sum()
    FP = np.logical_and(segmented, np.logical_not(reference)).sum()
    FN = np.logical_and(np.logical_not(segmented), reference).sum()

    precision = TP / (TP + FP) if (TP + FP) != 0 else 0
    recall = TP / (TP + FN) if (TP + FN) != 0 else 0
    f_measure = 2 * (precision * recall) / (precision + recall) if
(precision + recall) != 0 else 0

    return f_measure

# Primjer upotrebe
# Ovdje bi se učitala segmentirana slika i referentna slika
# i zatim pozvala funkcija calculate_f_measure

# f_measure_score = calculate_f_measure(segmented, reference)
# print(f"F-Measure: {f_measure_score:.4f}")
```

```

# Jaccard index

import numpy as np

def calculate_jaccard_index(segmented, reference):
    segmented_mask = segmented.astype(bool)
    reference_mask = reference.astype(bool)
    intersection = np.logical_and(segmented_mask, reference_mask)
    union = np.logical_or(segmented_mask, reference_mask)
    jaccard_index = intersection.sum() / union.sum()
    return jaccard_index

# Primjer upotrebe
# Ovdje bi se učitala segmentirana slika i referentna slika
# i zatim pozvala funkcija calculate_jaccard_index

# jaccard_index = calculate_jaccard_index(segmented, reference)
# print(f"Jaccard Index: {jaccard_index:.4f}")

```

### **# Variation of Information**

```

import numpy as np

def calculate_variation_of_information(segmented, reference):
    n = len(segmented)
    w = len(reference)

    hist_seg = np.histogram2d(segmented.ravel(), reference.ravel(),
bins=(n, w))[0]
    hist_seg = hist_seg / hist_seg.sum()

    prob_seg = hist_seg.sum(axis=1)
    prob_ref = hist_seg.sum(axis=0)

    h_seg = -np.sum(prob_seg * np.log(prob_seg + (prob_seg == 0)))
    h_ref = -np.sum(prob_ref * np.log(prob_ref + (prob_ref == 0)))

    vi = h_seg + h_ref - 2 * hist_seg[hist_seg > 0].sum()

    return vi

# Primjer upotrebe
# Ovdje bi se učitala segmentirana slika i referentna slika
# i zatim pozvala funkcija calculate_variation_of_information

# vi = calculate_variation_of_information(segmented, reference)
# print(f"Variation of Information: {vi:.4f}")

```

## Literatura

- [1] O. Alata and L. Quintard, “Is there a best color space for color image characterization or representation based on multivariate gaussian mixture model?” *Computer Vision and Image Understanding*, Volume 113, Issue 8, pp. 867-877, 2009.
- [2] M. Pagola, R. Ortiz, I. Irigoyen et al., “New method to assess barley nitrogen nutrition status based on image colour analysis: comparison with SPAD-502,” *Computers and Electronics in Agriculture*, Volume s65, No. 2, pp. 213–218, 2009.
- [3] J. M. Chaves-Gonzalez, M. A. Vega-Rodriguez, J. A. GomezPulido, and J. M. Sanchez-Perez, “Detecting skin in face recognition systems: a colour spaces study,” *Digital Signal Processing*, Volume 20, Issue 3, pp. 806-823, 2010.
- [4] A. Jurio, M. Pagola, M. Galar, C. Lopez-Molina, and D. Paternain, “A comparison study of different color spaces in clustering based image segmentation,” *Information Processing and Management of Uncertainty in Knowledge-Based Systems. Applications, vol. 81 of Communications in Computer and Information Science*, 13th International Conference, Dortmund, Germany, June 28 - July 2, pp. 532–541, 2010.
- [5] L. Busin, N. Vandebroucke, and L.Macaire, “Color spaces and image segmentation, ” *Advances in Imaging and Electron Physics*, Volume 151, pp. 65-168, 2008.
- [6] G. Ruiz-Ruiz, J. Gómez-Gil, and L. M. Navas-Gracia, “Testingdifferent color spaces based on hue for the environmentallyadaptive segmentation algorithm (EASA),” *Computers and Electronics in Agriculture*, Volume 68, No. 1, pp. 88-96, 2009.
- [7] F. Garcia-Lamont, J. Cervantes, A. Lopez-Chau, J. S. R. Castilla, “Color image segmentation using saturated RGB colors and decoupling the intensity from the hue”, *Multimedia Tools and Applications*, Volume 79, Issue 1-2, pp. 1555-1584, 2019.
- [8] V. Rajinikanth, M.S. Couceiro, “RGB Histogram based Color Image Segmentation Using Firefly Algorithm”, *Procedia Computer Science*, Volume 46, pp. 1449-1457, 2015.

- [9] D. D. Burdescu, M. Brezovan, E. Ganea, L. Stanescu, "A New Method for Segmentation of Images Represented in a HSV Color Space", *Advanced Concepts for Intelligent Vision Systems*, 11th International Conference, Bordeaux, France, September 28 - October 2, pp. 606–617. 2009.
- [10] N. Yadav, S. M. Alfayeed, A. Khamparia, B. Pandey, D. N. H. Thanh, S. Pande, "HSV model-based segmentation driven facial acne detection using deep learning", *Expert Systems*, Volume 39, Issue 3 2021.
- [11] V. Chernov, J. Alander, V. Bochko, "Integer-based accurate conversion between RGB and HSV color spaces", *Computers & Electrical Engineering*, Volume 46, pp. 328-337, 2015.
- [12] J. Ramírez, P. Temoche, and R. Carmona, "A volume segmentation approach based on GrabCut," *CLEI Electronic Journal*, Volume 3, No. 5, pp. 153-157, 2013.
- [13] Hmrishav Bandyopadhyay, An Introduction to Image Segmentation: Deep Learning vs. Traditional [+Examples], <https://www.v7labs.com/blog/image-segmentation>, posljednji put pristupano: 20. novembar 2023.
- [14] L. Shapiro, G Stockman, "Computer vision", *The University of Washington*, pp. 30-60, 2001.
- [15] Zack G. W, Rogers W. E, "Automatic measurement of sister chromatid exchange frequency". *Journal of Histochemistry & Cytochemistry*, pp. 741–753, 1977.
- [16] N. Otsu (1979). "A threshold selection method from gray-level histograms", IEEE Transactions on Systems, Man, and Cybernetics, 1979.
- [17] N. Sharma, "K-means Clustering Explained", <https://neptune.ai/blog/K-Means-clustering>, posljednji put pristupano: 20. novembar 2023.

- [18] H. P. Kriegel, E. Zimek, "The (black) art of runtime evaluation: Are we comparing algorithms or implementations?", *Knowledge and Information Systems*, pp. 341–378, 2016.
- [19] D. Khattab, H. M. Ebied, A.S. Hussein, M. F. Tolba, "Color Image Segmentation Based on Different Color Spaces Models Using Automatic GrabCut", *The Scientific World Journal*, Volume 2014, Article ID 126025, 2014.
- [20] N. K. Gill, A. Sharma, "Vehicle Detection from Satellite Images in Digital Image Processing", *International Journal of Computational Intelligence Research 13*, pp 697-705, 2017.
- [21] T. Sørensen, "A method of establishing groups of equal amplitude in plant sociology based on similarity of species and its application to analyses of the vegetation on Danish commons", *Royal Danish Society of Sciences*, pp. 1-34, 1948.
- [22] Lee R. Dice (1945). "Measures of the Amount of Ecologic Association Between Species", *Ecology*, pp. 297–302, 1945.
- [23] Y. Sasaki, "The truth of F-Measure", University of Manchester, pp. 1-5, 2007.
- [24] NC. Chung, B. Miasojedow, M. Startek, A. Gambin, "Jaccard/Tanimoto similarity test and estimation methods for biological presence-absence data", *BMC Bioinformatics*, pp. 644, 2009.
- [25] M. Meila, "Comparing Clustering", *University of Washington*, pp. 1-15, 2003.
- [26] Liao P-S., Chen T-S. and Chung, P-C., "A fast algorithm for multilevel thresholding", *Journal of Information Science and Engineering*, pp 713-727, 2001.