

Duboko učenje za segmentaciju različitih abdominalnih organa iz medicinskih slika

Lovretić, Luka

Master's thesis / Diplomski rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:200:174543>

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

Download date / Datum preuzimanja: **2025-04-01**

Repository / Repozitorij:

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I
INFORMACIJSKIH TEHNOLOGIJA OSIJEK**

Sveučilišni studij

**DUBOKO UČENJE ZA SEGMENTACIJU RAZLIČITIH
ABDOMINALNIH ORGANA IZ MEDICINSKIH SLIKA**

Diplomski rad

Luka Lovretić

Osijek, 2023.

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA OSIJEK**Obrazac D1: Obrazac za ocjenu diplomskog rada na sveučilišnom diplomskom studiju****Ocjena diplomskog rada na sveučilišnom diplomskom studiju**

Ime i prezime pristupnika:	Luka Lovretić
Studij, smjer:	Sveučilišni diplomski studij Računarstvo
Mat. br. pristupnika, god.	D-1221R, 07.10.2021.
JMBAG:	0165078883
Mentor:	prof. dr. sc. Irena Galić
Sumentor:	dr. sc. Marija Habijan
Sumentor iz tvrtke:	
Predsjednik Povjerenstva:	prof. dr. sc. Krešimir Nenadić
Član Povjerenstva 1:	dr. sc. Marija Habijan
Član Povjerenstva 2:	doc. dr. sc. Krešimir Romić
Naslov diplomskog rada:	Duboko učenje za segmentaciju različitih abdominalnih organa iz medicinskih slika
Znanstvena grana diplomskog rada:	Umjetna inteligencija (zn. polje računarstvo)
Zadatak diplomskog rada:	Istražiti i opisati način dobivanja slika pomoću računalne tomografije (CT) i magnetske rezonance (MRI) te istaknuti njihove najvažnije karakteristike. Istražiti i opisati kliničku pozadinu (klinička potreba segmentacije, opisati abdominalne organe). Objasniti teorijske osnove obrade medicinskih slika neuronskim mrežama, tipove i najčešće korištene modele. Opisati razlike, prednosti i nedostatke korištenja 2D podataka i 3D podataka prilikom učenja neuronskih mreža. Dati kratak pregled područja prethodnih istraživanja. Razviti sustav za segmentaciju različitih
Datum ocjene pismenog dijela diplomskog rada od strane mentora:	01.07.2024.
Ocjena pismenog dijela diplomskog rada od strane mentora:	Izvrstan (5)
Datum obrane diplomskog rada:	15. 7. 2024.
Ocjena usmenog dijela diplomskog rada (obrane):	Izvrstan (5)
Ukupna ocjena diplomskog rada:	Izvrstan (5)
Datum potvrde mentora o predaji konačne verzije diplomskog rada čime je pristupnik završio sveučilišni diplomski studij:	30.07.2024.



FERIT

FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

IZJAVA O IZVORNOSTI RADA

Osijek, 30.07.2024.

Ime i prezime Pristupnika:

Luka Lovrečić

Studij:

Sveučilišni diplomski studij Računarstvo

Mat. br. Pristupnika, godina upisa:

D-1221R, 07.10.2021.

Turnitin podudaranje [%]:

12

Ovom izjavom izjavljujem da je rad pod nazivom: **Duboko učenje za segmentaciju različitih abdominalnih organa iz medicinskih slika**

izrađen pod vodstvom mentora prof. dr. sc. Irena Galić

i sumentora dr. sc. Marija Habijan

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.

Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis pristupnika:

SADRŽAJ

1. UVOD	1
1.1. Zadatak diplomskog rada	2
2. TRENUTNE METODE ZA SEGMENTACIJU ABDOMINALNIH ORGANA	3
2.1. Segmentacija abdominalnih organa koristeći nnU-Net.....	3
2.2. Segmentacija abdominalnih organa koristeći rezidualni U-Net.....	4
2.3. Segmentacija abdominalnih organa koristeći <i>Organ-Attention Network (OAN-RC)</i> i <i>Statistical Fusion (SF)</i>	5
3. MEDICINSKA I TEHNOLOŠKA POZADINA	7
3.1. Medicinska pozadina	7
3.1.1. Slezena.....	7
3.1.2. Bubrezi	8
3.1.3. Jetra	9
3.2. Tehnološka pozadina	10
3.2.1. Računalna tomografija(CT).....	10
3.2.2. Magnetska rezonancija (MRI)	12
3.3. Formati zapisa medicinskih slika	14
3.3.1. Analyze.....	14
3.3.2. NIFTI.....	15
3.3.3. DICOM.....	15
3.3.4. MINC.....	16
4. KONVOLUCIJSKE NEURONSKE MREŽE.....	17
4.1. Princip rada konvolucijskih neuronskih mreža	17
4.2. Slojevi u konvolucijskim neuronskim mrežama	19
4.2.1. Konvolucijski sloj.....	19
4.2.2. Sloj sažimanja.....	21
4.2.3. Sloj slučajnog izostavljanja	22
4.2.4. Sloj normalizacije serije	23
4.2.5. Potpuno povezani sloj.....	24
5. PROGRAMSKO RJEŠENJE	26
5.1. U-Net arhitektura neuronske mreže.....	26
5.1.1. ResNet U-Net	27

5.1.2. Ugniježđeni U-Net (U-Net++).....	28
5.2. Korišteni skup podataka	29
5.3. Učitavanje podataka	29
5.4. Predobrada podataka	31
5.4.1. Predobrada slike	32
5.4.2. Predobrada maske.....	33
5.5. Dinamičko upravljanje podacima	34
5.6. Funkcija gubitka	35
5.6.1. Binary Cross Entropy	36
5.6.2. DICE koeficijent.....	36
6. Dobiveni rezultati	37
6.1. Predviđanja modela na validacijskom skupu.....	39
6.2. Predviđanja modela na testnom skupu.....	40
7. Zaključak	41
Literatura.....	42

1. UVOD

U današnje vrijeme svjedoci smo značajnog rasta različitih računalnih tehnologija koje se primjenjuju u raznim industrijama i sektorima, što rezultira povećanjem produktivnosti i pojednostavljenjem rada pojedinca. Napredak računalnih tehnologija osjeti se i u medicini. Posebno se osjeti u području radiologije, gdje se pomoću medicinskih snimanja poput ultrazvuka, računalne tomografije (CT), magnetske rezonancije (MRI) i slično sastavljaju 3D medicinske slike na temelju kojih se mogu definirati dijagnoze, terapije ili daljnja istraživanja. Snimljene medicinske slike mogu poslužiti kao podaci koji se predaju metodama dubokog učenja. U ovom diplomskom radu opisan će se teorijski i praktični dio primjene dubokog učenja za segmentaciju različitih abdominalnih organa iz multimodalnih medicinskih slika. Segmentacija se koristi za precizno definiranje pozicije, veličine i oblika organa ili drugih bioloških struktura unutar medicinske slike. Također je bitan čimbenik pri pronalasku interesnih regija u medicinskim slikama poput tumora, lezija, cista i slično. Automatska segmentacija može biti vrlo koristan i moćan alat koji radiolozima omogućuje precizniju i kvalitetniju interpretaciju medicinskih slika, pomaže u dijagnostici, planiranju liječenja i praćenja pacijenata. Rješenje segmentacije koristeći neuronske mreže ima i svoje izazove. Segmentacija medicinskih slika koristeći neuronske mreže težak je zadatak zbog različitih ograničenja nametnutih postupkom nabave medicinskih slika (zbog privatnosti i zaštite osjetljivih podataka o pacijentu) i bioloških varijacija među pacijentima.

Diplomski rad sastoji se od sedam glavnih poglavlja. U uvodnom dijelu opisan je kratak uvod u diplomski rad i opis zadatka diplomskog rada. U drugom poglavlju opisuju se trenutno značajne metode za segmentaciju abdominalnih organa i njihove osnovne karakteristike. U trećem poglavlju kratko su opisani svi abdominalni organi koji se tijekom praktičnog dijela segmentiraju iz medicinskih slika. Četvrto poglavlje predstavlja tehnološku pozadinu dobivanja slike iz medicinskih uređaja, gdje se opisuje proces dobivanja medicinskih slika računalnom tomografijom (CT) i magnetskom rezonancom (MRI) te su opisani i najpoznatiji formati za zapis medicinskih slika. Peto poglavlje opisuje svojstva konvolucijske neuronske mreže, njezinu svrhu, princip rada i osnovni oblik. Opisani su slojevi neurona koji čine osnovnu arhitekturu jedne konvolucijske neuronske mreže. U šestom poglavlju nalazi se programsko rješenje zadatka. Također se opisuje U-Net arhitektura i njene korištene inačice, skup podataka koji je korišten, analiziraju se dijelovi koda i vizualiziraju se dobiveni rezultati. U zadnjem poglavlju dan je sveukupni zaključak na temelju rezultata razvijenog sustava za segmentaciju abdominalnih organa.

1.1. Zadatak diplomskog rada

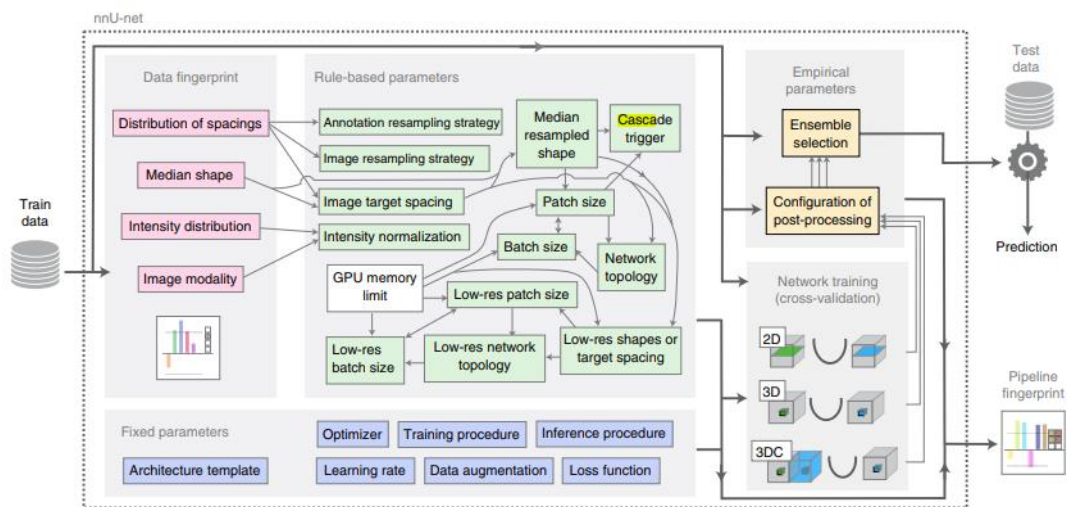
Zadatak diplomskog rada je istražiti i opisati način dobivanja slika pomoću računalne tomografije (CT) i magnetske rezonance (MRI) te istaknuti njihove najvažnije karakteristike. Potrebno je istražiti i opisati kliničku pozadinu (klinička potreba segmentacije i opisati abdominalne organe) te objasniti teorijske osnove obrade medicinskih slika neuronskim mrežama, tipove i najčešće korištene modele. U radu je potrebno opisati razlike, prednosti i nedostatke korištenja 2D podataka i 3D podataka prilikom učenja neuronskih mreža. Dati kratak pregled područja prethodnih istraživanja. Razviti sustav za segmentaciju različitih abdominalnih organa pomoću U-Net baziranih arhitektura neuronskih mreža. Prikazati i objasniti rezultate te odrediti preciznost izvođenja razvijenog sustava. Tehnologije koje će se koristiti su Python i PyTorch uz dodatne biblioteke za obradu i prikaz podataka.

2. TRENUTNE METODE ZA SEGMENTACIJU ABDOMINALNIH ORGANA

Automatska segmentacija abdominalnih organa složen je proces kojem je moguće pristupiti raznim načinima. Osim tradicionalnih pristupa rješavanju ovog problema poput klasičnih metoda obrade slika, postoje također pristupi u kojima se koriste metode dubokog učenja. Većina pristupa koji su izvedeni metodama dubokog učenja temeljeni su na U-Net arhitekturi. U-Net arhitektura jedna je od standardnih konvolucijskih neuronskih mreža (engl. *Convolutional Neural Network, CNN*) često upotrebljavana za klasifikaciju medicinskih slika. U ovom poglavlju prikazano je nekoliko metoda za segmentaciju abdominalnih organa koje su izvedene iz prethodnih istraživanja.

2.1. Segmentacija abdominalnih organa koristeći nnU-Net

G. Krishnan Murugesan, D. McCrumb i ostali [1] kreirali su nnU-Net arhitekturu. nnU-Net je samokonfigurirajuća arhitektura za segmentaciju medicinskih slika što znači da se navedena mreža sama konfigurira u smislu predobrade, arhitekture mreže, treniranja i naknadne obrade. Na ovaj način omogućen je automatizirani cjevovod (engl. *pipeline*) koji se može koristiti na bilo kojem skupu medicinskih podataka. nnU-Net generira tri U-Net konfiguracije: 2D U-Net, 3D U-Net koji radi na slikama u punoj rezoluciji i 3D U-Net kaskada u kojoj prvi model radi na slikama u smanjenoj rezoluciji, dok je drugi naučen za poboljšavanje segmentacijskih mapa koje prima od prethodnog modela u puno rezoluciji. nnU-Net odabire model koji je imao najbolje rezultate na temelju unakrsne provjere valjanosti (engl. *cross-validation*). Slika 2.1. prikazuje cijeli tijek rada nnU-Net arhitekture [2].



Sl. 2.1. Tijek rada nnU-Net arhitekture [2]

2.2. Segmentacija abdominalnih organa koristeći rezidualni U-Net

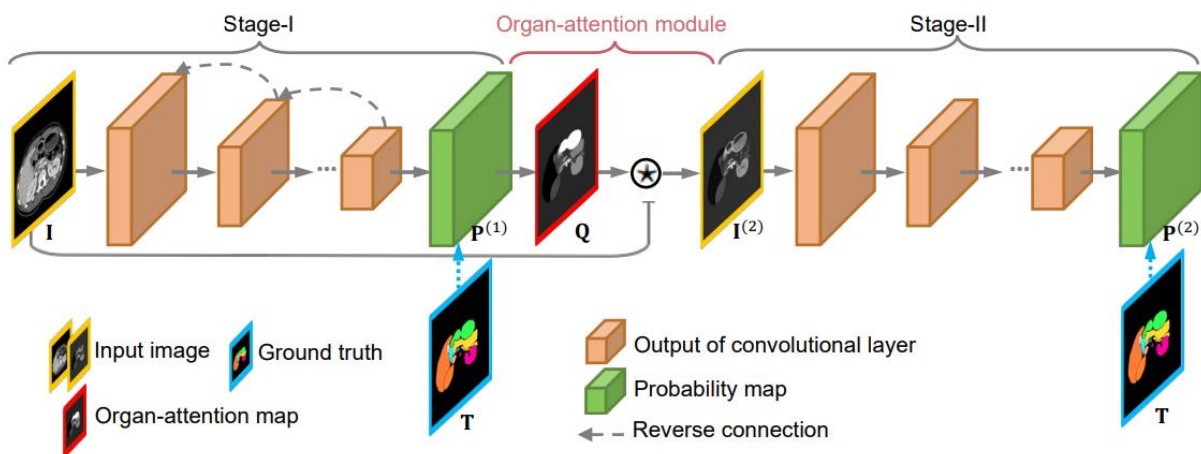
Satoshi Kondo i Satoshi Kasai [3] predložili su metodu koja se temelji na rezidualnoj U-Net arhitekturi koju su nazvali ResU-Net (engl. *Residual U-Net*). ResU-Net je varijanta osnovnog U-Net modela unutar kojeg se koriste rezidualni blokovi umjesto konvolucijskih. Rezidualni blok definiran je kao blok slojeva sa preskočnom vezom koji uči preko rezidualnih funkcija s referencom na ulazni sloj. S obzirom da u ovoj metodi model koristi 3D medicinske podatke, prilikom faze treniranja modela, nasumično se odabiru 3D volumeni iz ulaznih podataka (volumena) koji se koriste kao skup podataka za treniranje modela. Također je implementirana duboka supervizija (engl. *deep supervision*) za kalkulaciju gubitaka učenja (engl. *loss calculation*). Trenira se više modela. Svaki model samostalno se trenira nasumično izdvojenim kombinacijama skupova podataka za treniranje i validaciju. Model koji ima najmanje gubitke na validacijskom skupu uzima se kao završni model. Slika 2.2. prikazuje izgled ResU-Net arhitekture [4].



Sl. 2.2. 3D ResU-Net arhitektura [4]

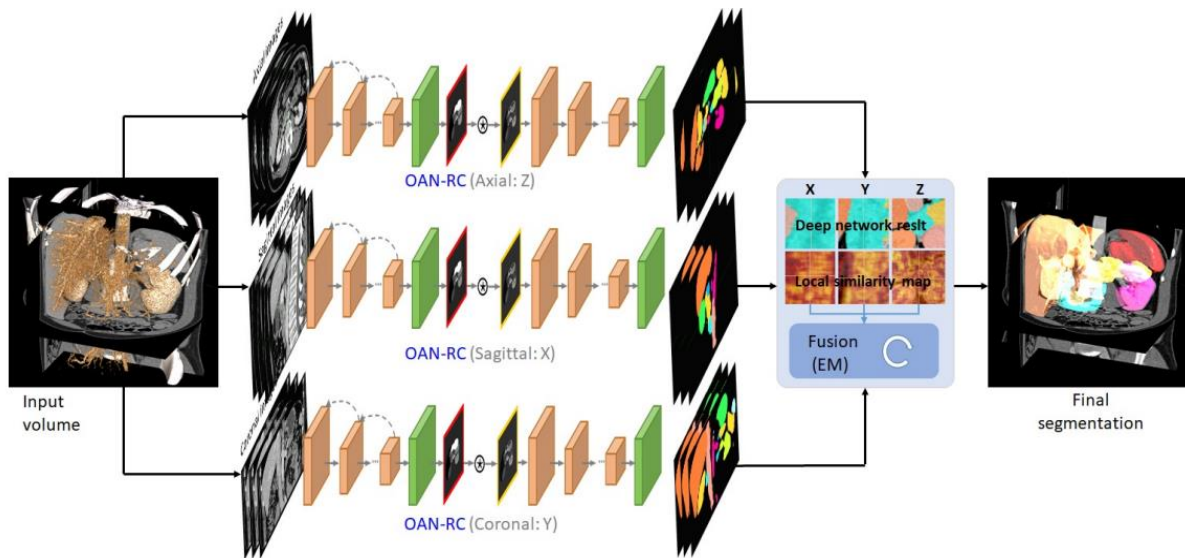
2.3. Segmentacija abdominalnih organa koristeći *Organ-Attention Network (OAN-RC)* i *Statistical Fusion (SF)*

Y. Wang, Y. Zhou, W. Shen i ostali [5] kreirali su *Organ Attention Networks – Reversed Connection* ili OAN-RC razvojni okvir koji služi za segmentaciju organa abdominalnog područja. OAN-RC je OAN (*Organ Attention Network*) duboka mreža sa implementiranom povratnom vezom koja je primjenjena na svim 2D pogledima (aksijalnim, sagitalnim i koronalnim) unutar 3D medicinske slike (volumena). OAN je duboka konvolucijska mreža koja se sastoji od dvije faze, prva i druga faza. Implementacijom povratne veze (engl. *Reverse Connection*) nižim slojevima mreže daje se više semantičkih podataka, čime se poboljšavaju rezultati treniranja. OAN sa implementiranom povratnom vezom naziva se *Organ Attention Network – Reversed Connection* ili OAN-RC. Prva faza (engl. *Stage I*) generira OAM (engl. *organ attention map*) iz mape vjerojatnosti (engl. *probability map*) za sve ulazne slike. Druga faza (engl. *Stage II*) spaja OAM generiran iz prve faze sa originalnom (ulaznom) slikom. Ova strategija u dvije faze smanjuje kompleksnost pozadine čime izdvaja ciljane abdominalne organe od ostatka slike. Na OAM iz prve faze dodaje se neuronska mreža koja minimizira sve nejasnoće i greške, što rezultira kvalitetnijom segmentacijom. Slika 2.3. prikazuje OAN-RC sa označenim fazama [5].



Sl. 2.3. Organ Attention Network – Reversed Connection(OAN-RC) [5]

Na samom kraju segmentacije koristi se statistička fuzija (engl. *Statistical Fusion*). Statistička fuzija smanjuje šumove na temelju lokalne strukturalne sličnosti (engl. *local structural similarity*), što poboljšava rezultate krajnje segmentacije. Slika 2.4. prikazuje izgled cijelog razvojnog okvira za segmentaciju abdominalnih organa [5].



Sl. 2.4. Izgled cijelog razvojnog okvira [5]

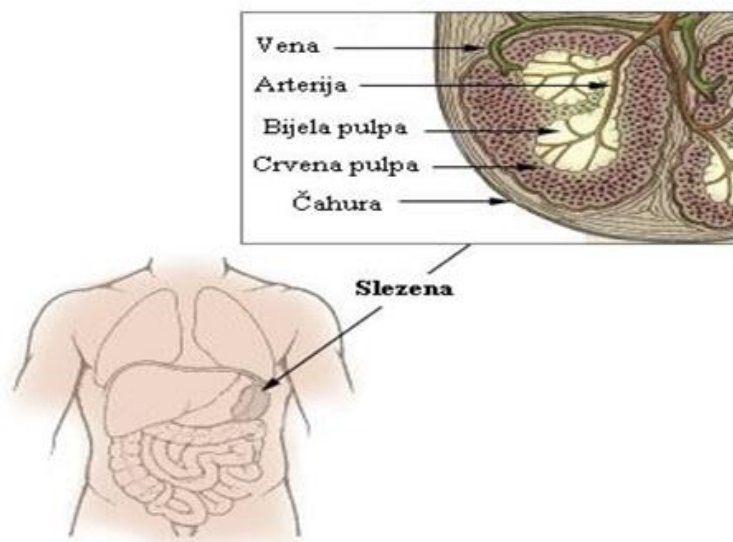
3. MEDICINSKA I TEHNOLOŠKA POZADINA

3.1. Medicinska pozadina

Unutar abdomena ili trbušne šupljine nalaze se vitalni organi probavnog sustava i organi poput bubrega i slezene. Zadatak diplomskog rada je segmentacija četiri abdominalna organa: slezene, desnog bubrega, lijevog bubrega i jetre. Slijedi kratak opis svakog od navednih abdominalnih organa.

3.1.1. Slezena

Slezena (engl. *spleen*) je organ koji se nalazi u tijelima svih kralježnjaka. Kod čovjeka se nalazi u trbuhu, točnije u lijevom dijelu trbušne šupljine, ispod rebara. Služi za razgradnju oštećenih crvenih krvnih stanica i kao spremnik krvi. Uklanja starije, oštećene ili nepotrebne krvne stanice. Također filtrira bakterije, viruse i druge strane čestice iz krvi. Sekundarna funkcija slezene je proizvodnja limfocita (bijelih krvnih stanica) koji služe za imunološku obranu tijela. Građena je od bijele i crvene pulpe i pripada limfnom sustavu. Slika 3.1. prikazuje sastav, položaj i oblik slezene [6].



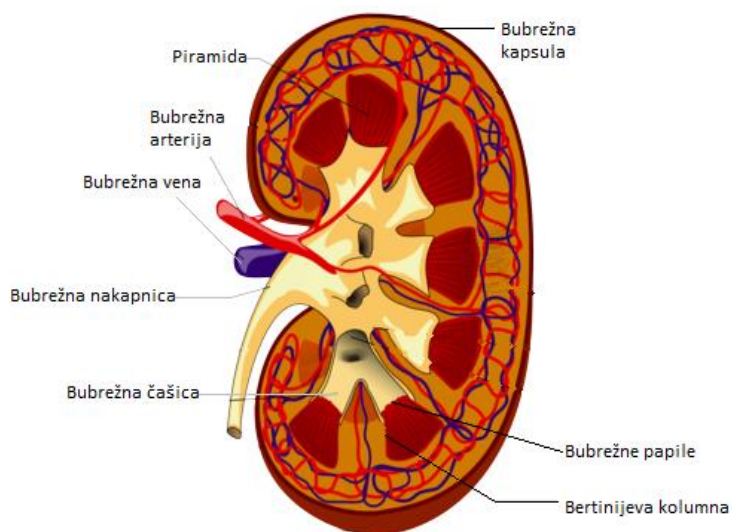
Sl. 3.1. Slezena i anatomija slezene [6]

3.1.2. Bubrezi

Bubrezi (engl. *kidneys*) su parni organi koji se nalaze u trbušnoj šupljini iza trbušnih organa. Desni bubreg je obično nešto niže od lijevog zbog položaja jetre, ali to može varirati ovisno o pojednicu. Bubrezi su odgovorni za filtriranje krvi, uklanjanje otpadnih tvari iz tijela, proizvodnju urina i održavanje ravnoteže tekućina i elektrolita u tijelu te reguliraju krvni tlak. Sastoje se od više dijelova, a najvažniji su: piramida, bubrežne krvne žile, bubrežna kapsula, bubrežna nakapnica, bubrežna čašica, bubrežne papile i Bertinijeva kolumna. Iako su funkcije oba bubrega vrlo slične, neke bolesti ili stanja mogu utjecati samo na jedan bubreg poput blokade urinarnog trakta, kamenja u bubregu i slično. Slike 3.2. [7] i 3.3. [8] prikazuju položaj bubrega u ljudskom tijelu i anatomiju jednog bubrega



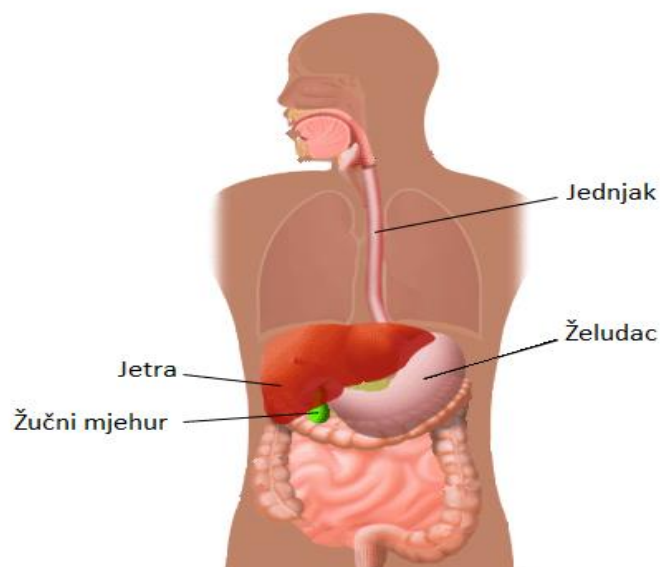
Sl. 3.2. Bubrezi u ljudskom tijelu [7]



Sl. 3.3. Anatomija bubrega [8]

3.1.3. Jetra

Jetra (engl. *liver*) jedan je od najvažnijih organa svakog kralježnjaka. Ima razne funkcije poput sinteze proteina i stvaranja biokemijskih molekula. Najveća je žlijezda u tijelu, a služi za skladištenje hranjivih tvari i neutraliziranje štetnih tvari. Kod čovjeka, jetra ima bitnu ulogu u metabolizmu obavljajući mnoštvo funkcija poput odlaganja glikogena i proizvodnju krvnih bjelancevina. Također proizvodi i žuč, tekućinu koja je bitna za probavu. Kod odraslih, jetra je mekan, smeđocrveni organ klinastog oblika. Krvožilni sustav jetre služi kao dodatni spremnik krvi u kojem se običajeno nalazi oko 500 mililitara krvi. Slika 3.4. prikazuje položaj jetre u ljudskom tijelu naspram okolnih organa i njen oblik [9].



Sl. 3.4. Jetra u ljudskom tijelu [9]

3.2. Tehnološka pozadina

Snimanje tijela CT i MRI uređajima najznačajnije su metode za dobivanje medicinskih slika u modernoj medicini. Dobivanje slike iz navedenih uređaja ključan je postupak za dijagnostiku i praćenje raznih bolesti. Ovi napredni medicinski uređaji omogućuju liječnicima dobivanje detaljnih slika tkiva, organa i kostiju, što pridonosi dijagnosticiranju i praćenju bolesti, te otkrivanju deformacija i anomalija unutar tijela. CT i MRI uređaji moraju biti korišteni isključivo od strane stručnog osoblja, poput radiologa, jer prilikom snimanja pacijent je izložen ionizirajućem zračenju ili snažnom magnetskom polju, ovisno koji uređaj se koristi. Važno je napomenuti da su ovi postupci obično provedeni samo kada postoje medicinski razlozi. Uz brži razvoj tehnologije skeniranja, uređaji postaju sve sofisticiraniji i precizniji i omogućavaju medicinskim stručnjacima dobivanje sve detaljnijih i kvalitetnijih slika. Poboljšanjem kvalitete medicinske slike unaprjeđuje se i kvaliteta rezultata raznih softverskih alata koji pružaju podršku medicinskim stručnjacima.

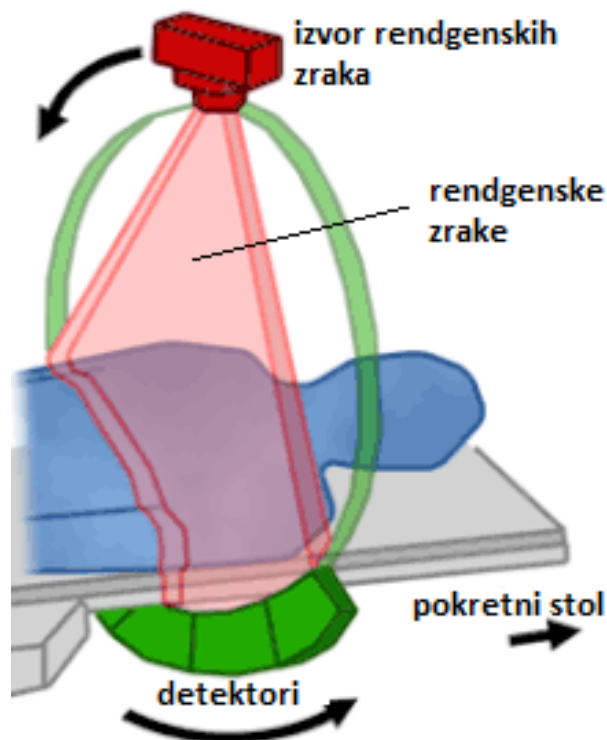
3.2.1. Računalna tomografija(CT)

Računalna tomografija ili CT (engl. *Computer Tomography*) medicinska je dijagnostička tehnika koja radi na principu rendgenskih zraka (engl. *X-ray*) i računalne obrade podataka kako bi stvorila detaljne trodimenzionalne slike unutrašnjosti tijela. CT skeniranje odvija se preko CT uređaja. Slika 3.5. prikazuje izgled CT uređaja [10].



Sl. 3.5. CT uređaj [10]

Izgled CT uređaja može varirati ovisno o proizvođaču, ali svi imaju sličan osnovni oblik. Uređaj ima veliku okruglu strukturu koja se naziva stalak ili *gantry*. *Gantry* je otvoren s obje strane kako bi omogućio prolazak pacijenta kroz njegovo središte. Na jednoj strani *gantry*-ja nalazi se rendgenska cijev, dok se na suprotnoj strani nalazi niz detektora. Pacijent se smješta na pokretni stol u ležećem položaju i prolazi kroz središte *gantry*-a. Računalni sustav obrađuje prikupljene podatke i generira slike. Snimanje se temelji na slabljenju ili atenuaciji rendgenskih zraka prolaskom kroz dio tijela koji se snima. Prilikom snimanja dolazi do apsorpcije rendgenskih zraka. Slabljenje rendgenskih zraka izražava se koeficijentom apsorpcije, a on ovisi o atomskom broju i elektronskoj gustoći tkiva te energiji rendgenskih zraka. Nakon prolaska kroz tijelo pacijenta, zrake ulaze u detektore koji ih pretvaraju u električne signale. Slika 3.6. prikazuje princip rada CT uređaja. Uz različite matematičke algoritme rekonstruira se slika snimanog objekta. Zdravstveni rizik pacijenta prilikom snimanja CT uređajem je izloženost određenoj dozi ionizirajućeg zračenja [11].



Sl. 3.6. Princip rada CT uređaja [11]

3.2.2. Magnetska rezonancija (MRI)

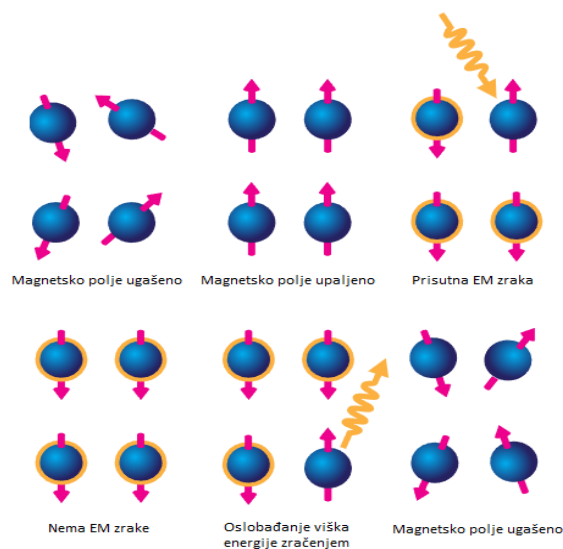
Magnetska rezonancija ili MRI (engl. *magnetic resonance imaging*), također poznata kao magnetska rezonantna tomografija. Jednako kao i računalna tomografija, magnetska rezonancija je radiološka tehnika snimanja koja se koristi za dobivanje detaljnih slika unutrašnje strukture tijela. Princip rada se temelji na nuklearnoj magnetskoj rezonanciji, tj. korištenju magnetskih polova i radiovalova. Nuklearna znači da se odnosi na jezgre atoma(nukleuse), magnetska jer se odvija u magnetskom polju, a rezonancija jer se iskorištava posebno svojstvo jezgri da apsorbiraju zračenje točno određene energije. Slike se dobivaju pomoću MRI uređaja koji izgleda slično kao CT uređaj, ali princip rada im je potpuno različit. Slika 3.7. prikazuje izgled MRI uređaja [12].



Sl. 3.7. MRI uređaj [12]

MRI uređaj se sastoji od magneta, izvora radiovalnog zračenja i detektora. Magnet proizvodi magnetsko polje određene jačine i u to magnetsko polje smješta se osoba koja se snima. Da bi bilo lakše objasniti kako se na temelju navedenih fizikalnih pojava dobije medicinska slika, potrebno je proći kroz nekakav uvod kako funkcionira fizika u pozadini cijelog procesa. Atomi se sastoje od tri čestice: protona(pozitivno nabijeni), elektrona(negativno nabijeni) i neutrona koji nisu nabijeni. Neutroni i protoni čine jezgru, a elektroni se nalaze u ljuskama koje okružuju jezgru. Na temelju količine svake čestice definiraju se kemijska svojstva elementa. Gibanjem tih čestica nastaje magnetsko polje. Jezgre atoma i molekula se rotiraju i samim time definiraju smjer magnetskog polja. Jezgre imaju rotaciju oko svoje osi koja može biti u bilo kojem smjeru. To

svojtvo naziva se spin. Spin definira smjer magnetskog polja. Pod utjecajem nekog vanjskog magnetskog polja, jezgre će se poredati tako da njihovo magnetsko polje bude ili paralelno(u smjeru) ili antiparalelno(suprotno) sa smjerom vanjskog magnetskog polja. Antiparalelan položaj zahtijeva više energije od paralelnog položaja, pa će ga i manji broj jezgri moći poprimiti. Razlika između paralelnog i antiparalelnog položaja ovisi o jakosti vanjskog magnetskog polja, tj. što je vanjsko magnetsko polje jače, razlika je veća. MRI uređaji koriste magnete koji proizvode magnetsko polje koje prisiljava jezgre da se poravnaju s tim poljem. Kada jezgre udari zraka elektromagnetskog zračenja(EM zraka) čija je energija točno jednaka razlici energija paralelnog i antiparalelnog stanja, to se naziva rezonancija. Tijekom rezonancije jezgre će preuzeti tu energiju i privremeno se okrenuti antiparalelno. Kad se elektromagnetska zraka isključi, jezgre se vraćaju u paralelan položaj i taj višak energije ponovo odašilju kao zračenje. Senzori MRI uređaja mogu očitati tu energiju koja se oslobađa. Vrijeme potrebno da se jezgre vrate na svoje položaje i količina oslobođene energije ovisna je o okruženju i kemijskoj prirodi molekula. Na temelju tih parametara moguće je razlikovati različite vrste tkiva i izvršiti računalnu interpretaciju, tj. učitati ih na računalo. MRI je vrlo sigurna dijagnostička metoda jer ne koristi ionizirajuće zračenje kao računalna tomografija, već se oslanja na snažno magnetsko polje i radiofrekvencijske impulse. Jedina potencijalna opasnost su prisutnost metalnih implantanata u tijelu pacijenta. U magnetskoj rezonanciji koriste se magneti, pa bi svi metalni implantanti bili pod utjecajem magnetskog polja što može imati vrlo štetne posljedice za pacijenta. Iz tog razloga je važno liječniku prijaviti sve prisutne metalne predmete u tijelu prije MRI pregleda. Također, tijekom postupka snimanja, MRI uređaj proizvodi veliku buku, pa je potrebna i odgovarajuća zaštita za uši. Slika 3.8. prikazuje ponašanje atoma pod utjecajem magnetskog polja i EM zraka [13].



Sl. 3.8. Ponašanje jezgre pod utjecajem mag. polja i EM zrake [13]

3.3. Formati zapisa medicinskih slika

Formati zapisa medicinskih slika su ključni za snimanje, prijenos i analizu medicinskih podataka. Omogućuju precizno vizualiziranje unutarnjih struktura tijela. Osmišljeni su s ciljem očuvanja kvalitete slike, sigurnosti podataka i kompatibilnosti među različitim medicinskim uređajima i softverima. Za razliku od običnih formata za slike poput JPEG-a, PNG-a i TIFF-a, medicinski formati slika imaju puno veću kvalitetu slike s velikom preciznošću na detaljima, jer su posebno dizajnirani da zadrže oštrinu slike, kontrast i detalje koji su bitni za dijagnozu i analizu medicinskih stanja pacijenata. Također sadrže metapodatke poput informacija o pacijentu, njegovom položaju tijekom snimanja, vrsti pregleda i drugim relevantnim podacima. Bitna značajka medicinskih formata je kompatibilnost. Omogućava da slike budu kompatibilne i čitljive na različitim platformama, što je važno za razmjenu slika među zdravstvenim ustanovama i timovima. Medicinski formati obično su usklađeni s propisima o sigurnosti i privatnosti zdravstvenih podataka. Uključuju funkcionalnost za enkripciju, autentifikaciju i kontrolu pristupa kako bi se očuvala zaštita osjetljivih medicinskih informacija. To je jedan od razloga zbog kojeg može biti teško pronaći dovoljan broj medicinskih podataka za treniranje modela dubokog učenja. Različiti formati zapisa medicinskih slika imaju svoje karakteristike, prednosti i ograničenja. U ovom odlomku prikazane su karakteristike četiri najpoznatija medicinska formata: NIfTI, DICOM, Analyze i MINC.

3.3.1. Analyze

Analyze je jedan od prvih medicinskih formata. Razvijen je od strane *Biomedical Imaging Resource* (BIR), američkog neprofitnog medicinskog centra. Pojavljuje se u osamdesetim godinama prošlog stoljeća. Dizajniran je za multidimenzionalne podatke poput 3D i 4D volumena. Bio je velik napredak za pohranu medicinskih slika koja je prije toga uglavnom podržavala samo 2D formate. Jedna Analyze datoteka sastojala se od dvije binarne datoteke: medicinsku sliku koja je imala „.img“ ekstenziju i zaglavlje sa „.hdr“ ekstenzijom, koje je sadržavalo metapodatke kao što je broj piksela, orijentacija slike, tip podatka i slično. Zaglavlje je fiksne duljine od 348 bajta.

3.3.2. NIfTI

NIfTI (engl. *Neuroimaging Informatics Technology Initiative*) je široko prihvaćen format zapisa medicinskih slika kojeg podržava velik broj programskih jezika i računalnih alata korištenih u medicini. Razvijen je kako bi poboljšao nedostatke prethodnog standarda za pohranu medicinskih slika, Analyze formata. Sastoji je od medicinske slike i zaglavlja koji su kombinirani u jednoj binarnoj datoteci. Svaka datoteka sadrži slikovne podatke koji predstavljaju volumetrijske informacije o medicinskim slikama. Unutar osnovnog NIfTI formata postoje NIfTI-1 i NIfTI-2 formati. Razlika između njih je to što je NIfTI-2 napravljen kao nadogradnja na NIfTI-1, može poprimiti veću količinu podataka. Ekstenzija NIfTI datoteke je „.nii“ ili „.nii.gz“ u slučaju kada je datoteka kompresirana.

3.3.3. DICOM

DICOM (engl. *Digital Imaging and Communication in Medicine*) je komunikacijski protokol i format datoteke za zapis medicinskih podataka. U sebi može sadržati informacije poput medicinske slike, razne informacije o pacijentu i slično. Format osigurava da svi podaci ostanu zajedno, kao i mogućnost prijenosa navedenih informacija između uređaja koji podržavaju DICOM format. DICOM je, poput NIfTI, format prihvaćen diljem svijeta. Od svoga predstavljanja 1993. godine, DICOM je revolucionirao radiološku praksu, zamjenjujući fizičku rendgensku sliku digitalnim zapisom. Najveća razlika između DICOM i NIfTI formata je to što, za razliku od NIfTI formata, DICOM 3D podatke interpretira kao skup 2D slojeva. Ovakav pristup omogućuje pregled bilo kojeg željenog 2D presjeka (*slice-a*) danog volumena, što može biti vrlo korisno pri analizi ljudskog tijela. Također, bitno je napomenuti da se NIfTI može pretvoriti u DICOM seriju (engl. *DICOM Series*), što daje mogućnost pregleda željenih presjeka volumena koji je prije toga bio spremljen u NIfTI formatu.

3.3.4. MINC

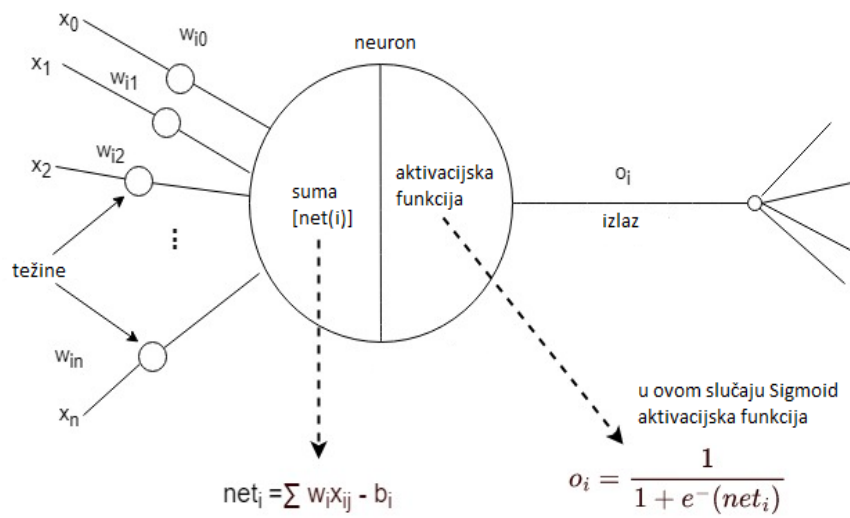
MINC (engl. Medical Imaging NetCDF) je format za medicinske slikovne podatke koji se temelji na NetCDF (engl. *Network Common Data Format*) standardu. MINC format datoteke i alati izvorno su osmišljeni, napisani i objavljeni od strane Petera Neelina 1992. godine. Razlog nastanka MINC formata je Neelinova frustracija zbog rukovanja s više medicinskih formata datoteka iz različitih skenera i istraživačkih grupa. MINC je posebno dizajniran kako bi medicinskoj zajednici pružio fleksibilan i proširiv format koji može upravljati širokim rasponom podataka proizvedenih modernim medicinskim uređajima. MINC datoteke često se koriste za prikaz 2D ili 3D medicinskih slikovnih podataka.

4. KONVOLUCIJSKE NEURONSKE MREŽE

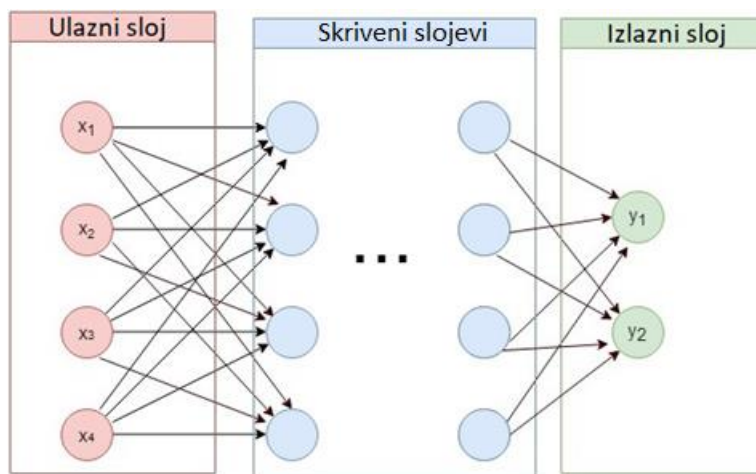
Konvolucijske neuronske mreže (engl. *Convolutional Neural Network*) ili CNN jedan su od najmoćnijih alata u području umjetne inteligencije i obrade slike. Ova vrsta neuronskih mreža posebno je dizajnirana za analizu vizualnih podataka i postigla je izvanredne rezultate u mnogim područjima kao što su prepoznavanje objekata, klasifikacija slika, detekcija uzoraka i segmentacija. Ono što čini konvolucijske neuronske mreže iznimno efikasnim je njihova sposobnost da nauče i izvlače relevantne značajke iz sirovih podataka. Umjesto da se ručno definiraju značajke, kao što je bilo u tradicionalnim metodama obrade slike, CNN-ovi sami uče optimalne značajke putem višestrukih slojeva filtriranja, konvolucije i agregacije. Konvolucijske neuronske mreže postale su ključni alat u raznim područjima, uključujući medicinu, robotiku, autonomna vozila, sigurnost i mnoge druge. Njihova sposobnost obrade vizualnih podataka i ekstrakcije bitnih informacija iz slika čini ih neizostavnima u području umjetne inteligencije.

4.1. Princip rada konvolucijskih neuronskih mreža

Konvolucijske neuronske mreže slične su uobičajenim neuronskim mrežama. Sastoje se od neurona koji imaju težine (engl. *weights*) i pristranost (engl. *bias*). U kontekstu neuronskih mreža, neuron je osnovna jedinica obrade inspiriran na načinu rada ljudskog mozga. Svaki neuron je matematička operacija koja svoje ulazne podatke množi sa svojom težinom, zbraja ih sa svojim promjenjivim *bias*-om i tu vrijednost predaje određenoj aktivacijskoj funkciji. Aktivacijska funkcija je nelinearna funkcija, poput Sigmoid ili ReLu, koja kao ulaz prima linearne podatke, a kao izlaz vraća nelinearne podatke. Izlazni podaci iz aktivacijske funkcije prenose se svim neuronima u idućem sloju. Na slici 4.1. prikazan je princip rada neurona. Svaka neuronska mreža sastoji se od slojeva koji rade neku vrstu transformacije i rezultate šalju dalje sljedećem sloju [14]. Jedan sloj mreže sastoji se od više neurona, čime se mreža može podijeliti na tri dijela: ulazni sloj, skriveni slojevi i izlazni sloj. Ulazni sloj prvi je sloj u mreži koji prima podatke iz vanjskog svijeta i koje prenosi dalje skrivenom sloju na obradu. Ulazni sloj početak je tijeka rada svake neuronske mreže. Neuronska mreža može imati samo jedan ulazni i jedan izlazni sloj. Izlazni sloj je zadnji sloj u neuronskoj mreži čija aktivacijska funkcija daje krajnji rezultat (predikciju) mreže. Neuronska mreža može imati više skrivenih slojeva koji primaju ulazne podatke od ulaznog sloja, obrađuju ih i na kraju predaju izlaznom sloju. Slika 4.2. prikazuje tri sloja u neuronskim mrežama [14].



Sl. 4.1. Princip rada neurona [14]



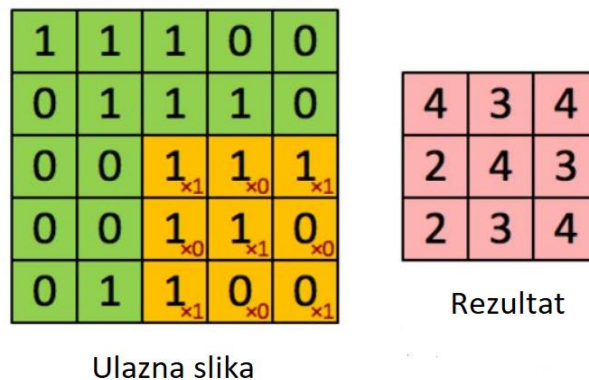
Sl. 4.2. Prikaz slojeva u neuronskim mrežama [14]

4.2. Slojevi u konvolucijskim neuronskim mrežama

Kako je ranije objašnjeno, sve neuronske mreže sastoje se od slojeva. U konvolucijskim neuronskim mrežama najkorišteniji su konvolucijski sloj (engl. *Convolution layer*), sloj sažimanja (engl. *Pooling layer*), sloj slučajnog izostavljanja (engl. *Dropout layer*), sloj normalizacije serije (engl. *Batch Normalization layer*) i potpuno povezani sloj (engl. *Fully Connected layer*).

4.2.1. Konvolucijski sloj

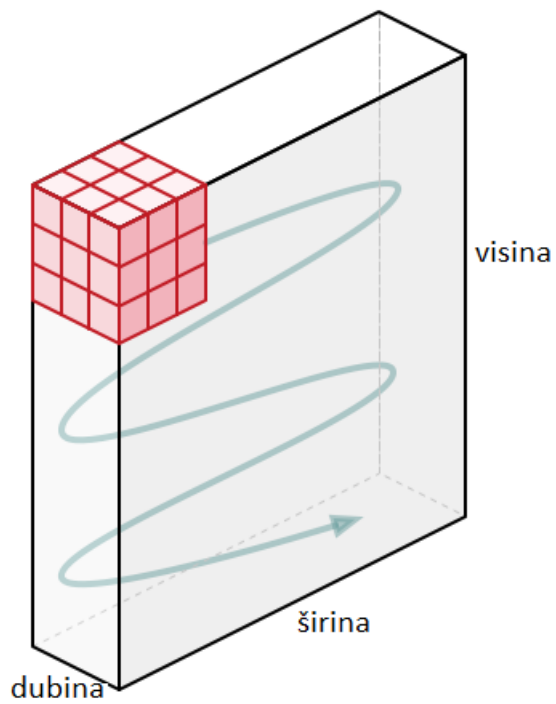
Konvolucijski sloj je najkorišteniji sloj u konvolucijskim neuronskim mrežama. U konvolucijskom sloju odvija se operacija konvolucije. Slika koju konvolucijski sloj prima je matrica u kojoj svaki element predstavlja jedan piksel (engl. *pixel*), a svaki piksel ima svoju vrijednost koja ovisi o njegovoj boji, dubini i drugim faktorima. Konvolucijski sloj koristi skup filtrirajućih jezgri ili *kernel* kako bi izvršio konvoluciju nad ulaznim podacima. *Kernel*, također poznat kao i filter, je proizvoljna matrica manjih dimenzija u usporedbi s ulaznom slikom koja se pomjera preko cijele ulazne slike izvršavajući operaciju konvolucije nad vrijednostima piksela slike. Rezultati konvolucije slažu se u novu sliku koja se predaje aktivacijskoj funkciji sloja. Slika 4.3. prikazuje rad konvolucijskog sloja, gdje je žutom bojom označen *kernel*, dok slika 4.4. i slika 4.5. prikazuju *kernel*, njegove vrijednosti i njegovo kretanje kroz sliku [15]. Cilj konvolucijskog sloja je dobivanje visokorazinskih značajki iz ulaznih slika. Izlaz zadnjeg konvolucijskog sloja pretvara se u 1D niz koristeći sloj izravnavanja (engl. *Flatten layer*), od kojeg počinje proces klasifikacije



Sl. 4.3. Konvolucija slike [15]

K =		
1	0	1
0	1	0
1	0	1

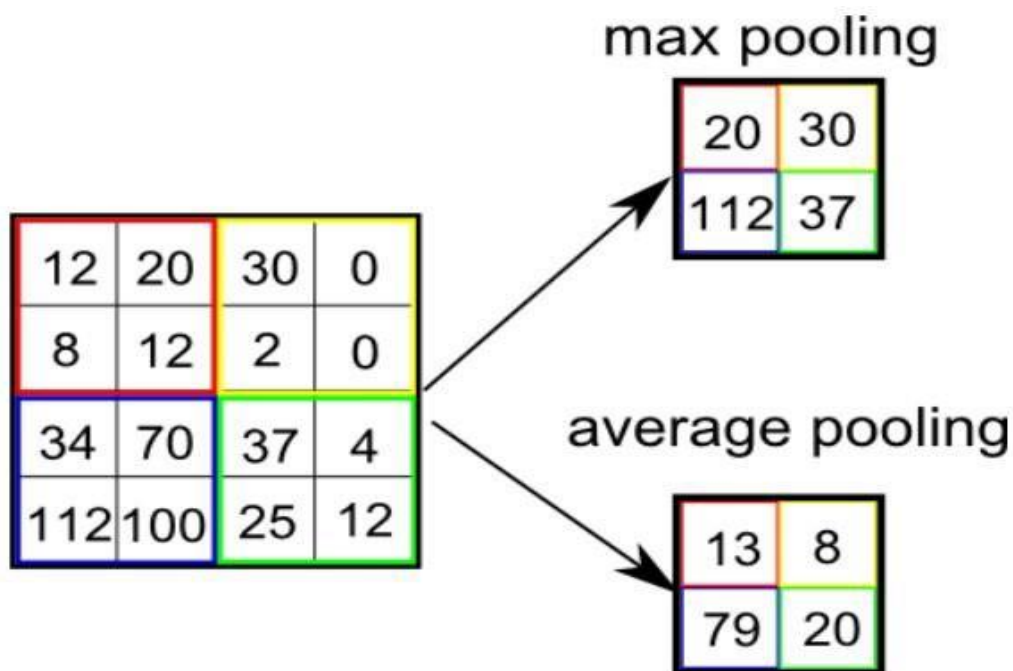
Sl. 4.4. *Kernel* korišten na primjeru prethodne konvolucije [15]



Sl. 4.5. Kretanje *kernela* kroz sliku [15]

4.2.2. Sloj sažimanja

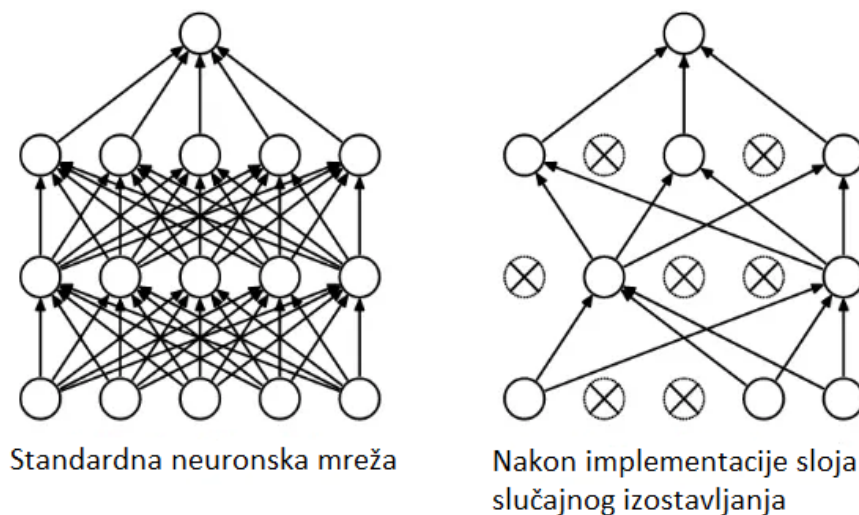
Sloj sažimanja (engl. *pooling layer*) radi na sličnom principu kao i konvolucijski sloj. Sloj sažimanja također ima matricu određenih dimenzija koje moraju biti manje od dimenzija ulazne slike. Matrica se određenim korakom kreće preko cijele ulazne slike, sprema u sebe njene vrijednosti i izvlači najveće (engl. *max pooling*) ili prosječne (engl. *average pooling*) vrijednosti unutar preklapljenih matrica u novu sliku. Ovim procesom kao izlaz se dobije ulazna slika smanjene dimenzionalnosti. Slika 4.6. prikazuje 2×2 *max pooling* i 2×2 *average pooling* [16]. Smanjenje dimenzionalnosti omogućava smanjenje broja parametara, što olakšava treniranje modela. Također doprinosi izvlačenju najvažnijih značajki iz ulaznih podataka. Međutim, treba biti oprezan prilikom odabira veličine matrice i koraka sažimanja jer preveliko sažimanje može dovesti do gubitka važnih informacija. Ukratko, sloj sažimanja smanjuje potrebnu računalnu snagu za obradu podataka putem smanjenja dimenzionalnosti.



Sl. 4.6. Princip rada *max pooling*-a i *average pooling*-a [16]

4.2.3. Sloj slučajnog izostavljanja

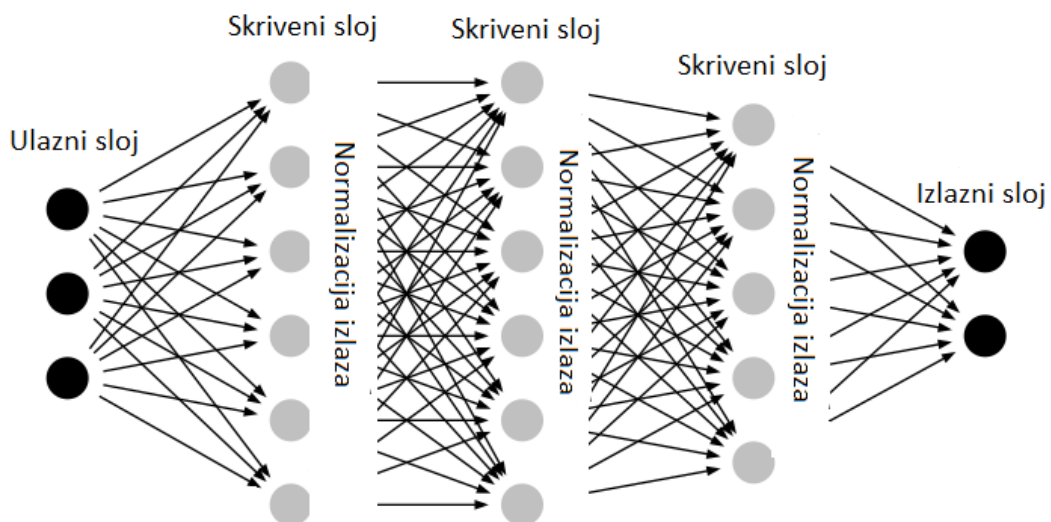
Neuronske mreže se vrlo lako prenauče (engl. *overfit*) nakon izvlačenja samo par primjera iz skupa podataka. *Overfitting* je nepoželjno ponašanje modela strojnog učenja kada model daje točna predviđanja samo za podatke koji su korišteni za treniranje istog tog modela, ali ne i za nove podatke. Zbog ove pojave koriste se slojevi slučajnog izostavljanja (engl. *dropout layer*). Slojevi slučajnog izostavljanja slučajno isključuju(deaktiviraju) neurone s vjerojatnošću p , gdje p predstavlja vjerojatnost isključivanja. To znači da tijekom svakog prolaza prema naprijed tijekom treniranja, neki neuroni neće doprinosti izračunu. Korištenjem ovog sloja smanjuje se međusobna ovisnost među neuronima, sprječava prenaučivanje i jača otpornost modela na šumove koji se nalaze u podacima. Može se također reći da treniraju nasumične podmreže glavne neuronske mreže. Bitno je napomenuti da se ovi slojevi koriste samo u fazi treniranja. Slika 4.7. prikazuje razliku među slojevima u neuronskoj mreži kada je implementiran sloj slučajnog isključivanja i kada nije [17].



Sl. 4.7. Implementacija sloja slučajnog izostavljanja u neuronsku mrežu [17]

4.2.4. Sloj normalizacije serije

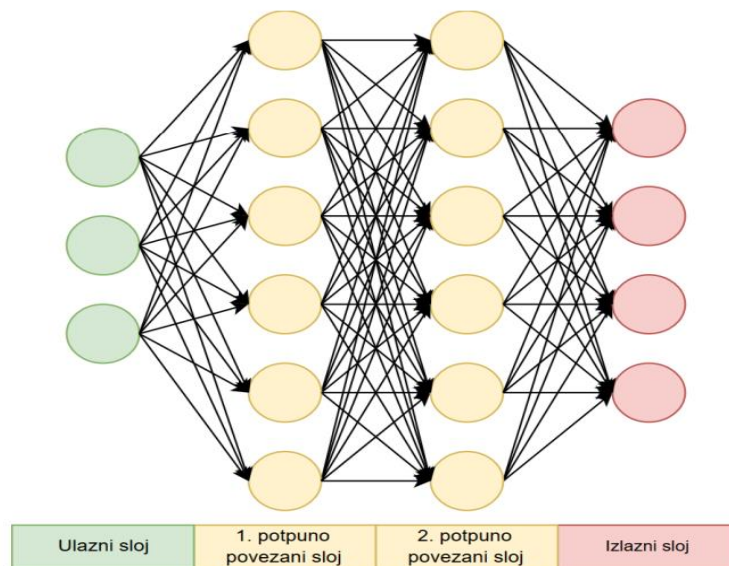
Sloj normalizacije serije je sloj koji se nalazi među skrivenim slojevima unutar neuronske mreže. Služi za normalizaciju izlaza jednog skrivenog sloja i prijenos normaliziranih podataka drugom skrivenom sloju. Sloj normalizacije serije računanjem srednje vrijednosti i varijance izlaza i primjenom linearnog skaliranja i pomaka normalizira primljene podatke od strane skrivenog sloja. Glavni razlog implementacije sloja normalizacije serije je poboljšanje stabilnosti i brzine učenja. Normalizacija se koristi kako bi se stabilizirala raspodjela vrijednosti izlaza i ublažili problemi nestajućeg gradijenta. Važno je napomenuti da ovaj sloj također ima regulacijski učinak koji može djelomično zamijeniti tehnike poput sloja slučajnog izostavljanja. Koristi se u mnogim arhitekturama neuronskih mreža i često se koristi u raznim zadacima strojnog učenja. Slika 4.8. prikazuje gdje bude smješten sloj normalizacije serije [18].



Sl. 4.8. Slojevi normalizacije serije u neuronskoj mreži [18]

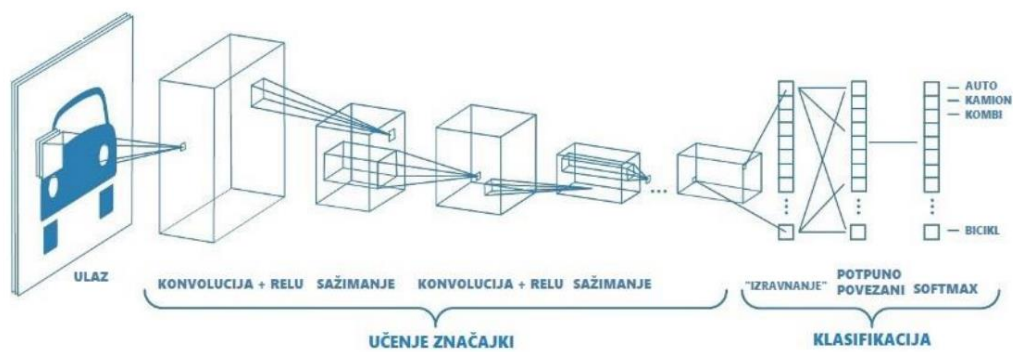
4.2.5. Potpuno povezani sloj

Potpuno povezani sloj (engl. *Fully-connected layer* ili *Dense layer*) igra ključnu ulogu u neuronskim mrežama. U ovom sloju, svaki neuron je povezan s neuronima iz prethodnog sloja, omogućavajući protok informacija na svakoj neuronskoj jedinici. Ulazni podaci u potpuno povezani sloj su izlazni podaci iz prethodnog sloja, dok su izlazni podaci potpuno povezanog sloja ponderirana suma tih ulaza. Tijekom faze učenja, težine se prilagođavaju kako bi se optimizirale performanse neuronske mreže. Nakon izračuna ponderirane sume, primjenjuje se aktivacijska funkcija koja unosi nelinearnost u izlaz sloja. Potpuno povezani slojevi uglavnom se koriste u zadnjem dijelu neuronskih mreža za zadatke poput klasifikacije ili regresije. Slika 4.9. prikazuje neuronsku mrežu s dva potpuno povezana sloja i njihov odnos s ulaznim i izlaznim slojem [19].



Sl. 4.9. Neuronska mreža sa dva potpuno povezana sloja [19]

Slika 4.10. prikazuje osnovni prikaz jedne konvolucijske neuronske mreže [20]. U segmentu izdvajanja značajki iz slika, sastoji se od dva konvolucijska sloja sa ReLU aktivacijskom funkcijom i dva sloja sažimanja. U klasifikacijskom dijelu neuronske mreže, izlaz drugog sloja sažimanja pretvara se u 1D niz i prenosi se u potpuno povezani sloj koji koristi softmax aktivacijsku funkciju. Izlazni sloj daje nam krajnje rezultate klasifikacije. U ovom slučaju radi se klasifikacija vozila, tj. prepoznavanje vozila sa slike.



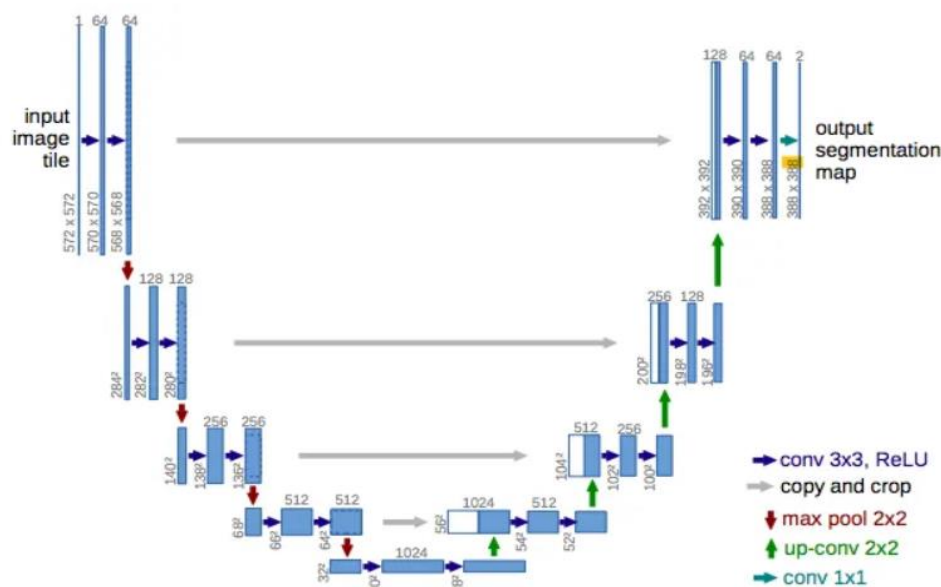
Sl. 4.10. Konvolucijska neuronska mreža [20]

5. PROGRAMSKO RJEŠENJE

Programsko rješenje izvedeno je u programskom jeziku Python. Za realizaciju svih akcija vezanih za stvaranje neuronske mreže, predobrade podataka, treniranje modela i njegovog vrednovanja korišten je PyTorch [21]. PyTorch je besplatna softverska biblioteka otvorenog koda (engl. *open source*) za strojno učenje i umjetnu inteligenciju. Ima poseban fokus na obuku i stvaranje dubokih neuronskih mreža. Također je korištena i NumPy biblioteka za lakši rad sa podacima. Skup podataka nad kojim se vršilo treniranje i testiranje modela je CHAOS i treniran je na dubokim neuronskim mrežama temeljenim na U-Net arhitekturi. Za treniranje modela korištene su inačice U-Net arhitekture ResNet U-Net i U-Net++.

5.1. U-Net arhitektura neuronske mreže

U-Net je popularna arhitektura neuronskih mreža koja je prvotno razvijena za zadatak semantičkog segmentiranja biomedicinskih slika, ali se kasnije proširila na druge primjene u računalnom vidu i obradi slike. Naziv „U-Net“ dobila je zbog svog oblika koje liči na slovo „U“. Sastoji se od dva simetrična dijela: područja sažimanja (engl. *contracting path* ili *downsampling path*) i područja širenja (engl. *expanding path* ili *upsampling path*). Slika 6.1. prikazuje izgled U-Net arhitekture [22]. Lijeva strana predstavlja područje sažimanja, dok desna područje širenja.

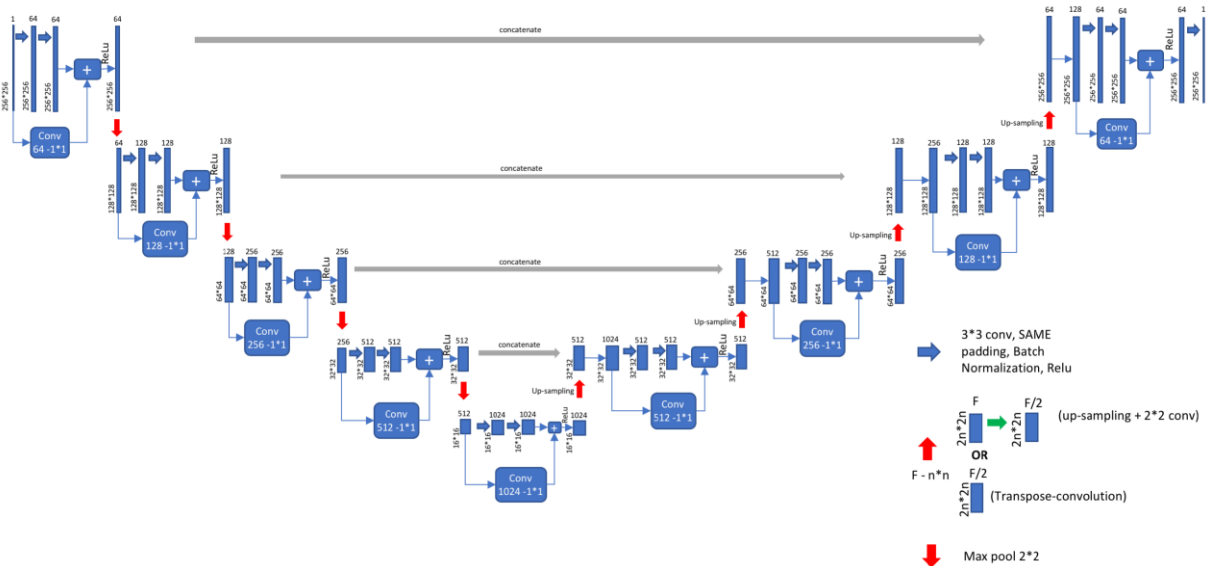


Sl. 6.1. U-Net arhitektura [22]

Područje sažimanja sastoji se od niza konvolucijskih slojeva koji postepeno smanjuju rezoluciju slike izvlačeći ključne značajke iz slike, čime omogućava mreži učenje različitih aspekta slike, poput kontura i oblika objekata. S druge strane, područje širenja sastoji se od slojeva koji povećavaju rezoluciju slike i omogućavaju rekonstrukciju detalja. Ovi slojevi koriste tehnike kao što su konvolucijski transponirani slojevi kako bi precizno rekonstruirali sliku na višoj rezoluciji. Povezanost između područja širenja i područja sažimanja osigurava da se bitni detalji sačuvaju prilikom rekonstrukcije, što je ključno za postizanje preciznih rezultata segmentacije.

5.1.1. ResNet U-Net

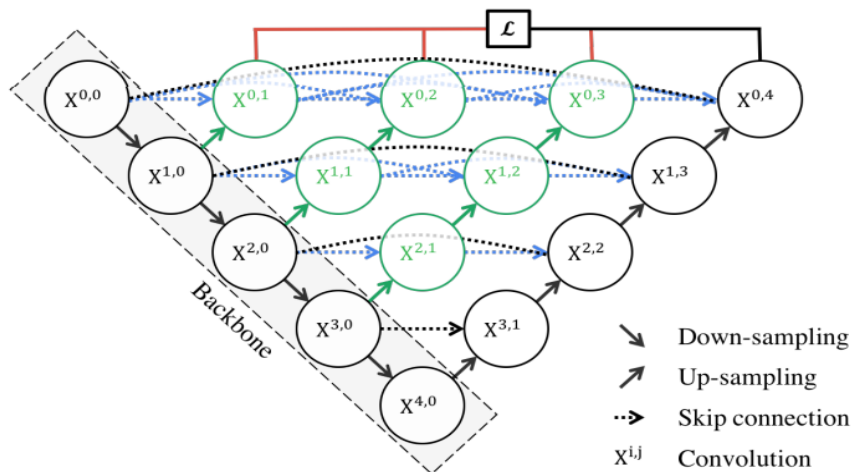
ResNet U-Net [3] je verzija U-Net-a u koja uključuje rezidualne blokove. Rezidualni blokovi u neuronskim mrežama omogućuju da bolje uče i zadržavaju informacije. Smanjuju problem nestajanja gradijenta, što omogućuje lakše treniranje modela. Povećavaju brzinu učenja i poboljšavaju rezultate modela. Svaki rezidualni blok sadrži nekoliko slojeva s dodatnom preskočnom vezom koja preskače jedan ili više slojeva unutar konvolucijskog bloka te povezuje ulaz bloka direktno s njegovim izlazom. Slika 6.2. prikazuje ResNet U-Net arhitekturu [23].



Sl. 6.2. Arhitektura ResNet U-Net modela [23]

5.1.2. Ugniježđeni U-Net (U-Net++)

Ugniježđeni U-Net (engl. *Nested U-Net*) unaprijeđena je verzija originalne U-Net arhitekture. Dizajnirana je kako bi se poboljšala preciznost segmentacije, posebno kada su granice objekata kompleksne ili nejasne. Ugniježđeni U-Net je gusta mreža preskočnih veza između enkoderskih i dekoderskih slojeva što omogućava bolji prijenos informacija i detaljniju rekonstrukciju slike. Dodatni konvolucijski slojevi poboljšavaju preciznost segmentacije. Slika 6.3. prikazuje arhitekturu ugniježđenog U-Net-a [24], gdje je crnom bojom označena originalna U-Net arhitektura, zelenom bojom dodatni konvolucijski blokovi i plavom bojom njihove poveznice. Crvenom bojom označena je duboka supervizija (engl. *deep supervision*) za praćenje gubitka nakon svakog dodatnog konvolucijskog bloka.



Sl. 6.3. Arhitektura U-Net++ modela [24]

5.2. Korišteni skup podataka

Za treniranje i testiranje modela korišten je *Combined Healthy Abdominal Organ Segmentation* ili skraćeno „CHAOS“ [25] skup podataka. CHAOS skup podataka sastoji se od CT i MRI slika različitih pacijenata. Dio skupa podataka koji predstavlja CT slike pacijenata sastoji se od dvadeset pacijenata za treniranje i dvadeset pacijenata za testiranje. Dio skupa podataka koji predstavlja MRI slike pacijenata također se sastoji od dvadeset pacijenata za treniranje i dvadeset pacijenata za testiranje sa T1-Dual i T2 SPIR sekvencama. U oba dijela skupa podataka slike za treniranje predstavljene su u DICOM formatu, dok su njihove maske u *.png* formatu. Skup za testiranje sadrži samo slike u DICOM formatu. U MRI skupu podataka unutar maski označena su sva četiri organa, dok u CT skupu označena je samo jetra.

5.3. Učitavanje podataka

Skup podataka definiran je koristeći *Dataset* klasu koja je sastavni dio PyTorch biblioteke. Ona omogućava strukturiranje i organizaciju podataka na način koji je prikladan za rad s modelima strojnog i dubokog učenja. Jedna od najvećih prednosti *Dataset* klase je prilagodljivost, što omogućava implementaciju specifične funkcionalnosti ovisno o strukturi i veličini skupa podataka. Slika 6.4. prikazuje konstruktor korisničke *ChaosDataset* klase koja je naslijeđena iz *Dataset* klase.

```
class ChaosDataset(Dataset):  
  
    def __init__(self, transform=None, transform_Y=None, dataset_type='train'):  
  
        self.data_type = dataset_type  
  
        self.dcm_paths = [Path(dcm) for dcm in sorted(list(glob.glob('**/*.dcm', recursive=True))  
                                                    if get_y_Path(Path(dcm)).exists())]  
  
        val_fraction = 0.3  
        if (dataset_type == 'valid'):  
            self.dcm_paths = self.dcm_paths[:int(len(self.dcm_paths)*val_fraction)]  
        else:  
            self.dcm_paths = self.dcm_paths[int(len(self.dcm_paths)*val_fraction):]  
  
        self.label_paths = [get_y_Path(dcm) for dcm in self.dcm_paths]
```

Sl. 6.4. Konstruktor *ChaosDataset* klase

Kod u konstruktoru prikazuje učitavanje putanja (engl. *path*) svih slika, te njihovo razdvajanje na skup za treniranje i skup za validaciju. Prilikom instanciranja klase, potrebno je konstruktoru predati koji se tip skupa podataka instancira, tj. radi li se o skupu za treniranje ili validaciju. Ukoliko se želi instancirati skup za validaciju modela, vrijednost parametra *dataset_type* mora biti string „valid“. Zadana vrijednost navedenog parametra je „train“, što znači da se radi o skupu za treniranje ukoliko se ne preda nikakva vrijednost. Klasa sadrži tri atributa: *data_type*, *dcm_paths* i *label_paths*. Atribut *data_type* string je koji predstavlja vrstu skupa podataka koji može biti za treniranje (vrijednost „train“) ili validaciju (vrijednost „valid“). Atribut *dcm_paths* predstavlja listu putanja do svake DICOM slike. Prilikom učitavanja *dcm_paths* atributa, implementirana je pomoćna funkcija *get_y_Path* koja provjerava postoji li za određenu putanju do DICOM slike i njena pripadajuća maska, čime se osigurava da su odabrane samo slike koje imaju svoje pripadajuće maske. Nakon učitavanja putanja, *dcm_paths* ima svoju vrstu obrade, gdje mijenja svoju veličinu ovisno radi li se skupu podataka za treniranje ili za validaciju. Ukoliko se radi o skupu za validaciju, uzima prvih trideset posto putanja za validaciju, dok u suprotnom slučaju uzima preostalih sedamdeset posto putanja kao skup za treniranje (isključujući prvih trideset posto putanja). Nakon definiranja *dcm_paths* atributa, definira se *label_paths* atribut koji sadrži putanje do maske za svaku DICOM putanju unutar *dcm_paths* liste.

5.4. Predobrada podataka

Predobrada (engl. *preprocessing*) podataka bitan je proces prije samog treniranja modela jer poboljšava točnost i učinkovitost istog modela. U slučajevima gdje se koriste konvolucijske neuronske mreže, slike se obrađuju koracima predobrade poput normalizacije, skaliranja, rezanja i augmentacije, te se nakon toga predaju modelu. Predobrada podataka izvodi se unutar *ChaosDataset* klase unutar *x_y_transforms* metode. Metoda kao parametre prima putanju do originalne DICOM slike i njene odgovarajuće maske i očekuje ih oboje kao NumPy nizove. Prvi dio predobrade jednak je i za DICOM sliku i za masku, dok se kasniji procesi preobrade razlikuju. Slika 6.5. prikazuje zajednički dio predobrade za slike i maske u *x_y_transforms* metodi.

```
def x_y_transforms(self, image: np.array = None, mask: np.array = None):  
  
    to_pil = transforms.ToPILImage()  
    image, mask = to_pil(image), to_pil(mask)  
  
    resize = transforms.Resize((128, 128), interpolation=Image.BICUBIC)  
    image = resize(image)  
    mask_resize = transforms.Resize((128, 128), interpolation=Image.NEAREST)  
    mask = mask_resize(mask)  
  
    centerCrop = transforms.CenterCrop(96)  
    image, mask = centerCrop(image), centerCrop(mask)  
  
    to_tensor = transforms.ToTensor()  
    image, mask = to_tensor(image), to_tensor(mask)
```

Sl. 6.5. *x_y_transforms* metoda

Parametri *image* i *mask* najprije se učitavaju preko *transforms* biblioteke [26] koja je izvedena iz TorchVision biblioteke. Koristi se *ToPILImage* klasa, koja sliku i masku pretvara u instance navedene klase, te se tako NumPy nizovi pretvaraju u PIL(*Python Imaging Library*) instance. Nakon instanciranja, slika i maska postaju kompatibilne sa klasama za predobradu iz *transforms* biblioteke. Za pravilan prijenos slike i maske konvolucijskoj neuronskoj mreži, potrebno je osigurati da sve slike i maske imaju jednake dimenzije. U tu svrhu koristi se klasa *Resize* koja prilagođava dimenzije slike i maske na 128x128 piksela. Nakon što je osigurano da su slika i maska jednakih dimenzija, slike su izrezane pomoću klase *CenterCrop*. *CenterCrop* je klasa za centrirano obrezivanje slika i slike su izrezane na dimenzije 96x96 piksela kako bi se uklonila što

veća količina pozadine i kako bi veći fokus bio na piksele koji predstavljaju ljudsko tkivo. Nakon promjena dimenzija i rezanja, slika i maska pretvorene su u *tensor*-e koristeći *ToTensor* klasu, čime su ujedno i normalizirane unutrašnjom normalizacijom *ToTensor* klase.

5.4.1. Predobrada slike

Nakon zajedničke predobrade, slijedi predobrada slike. Slika 6.6. prikazuje dio koda unutar *x_y_transforms* metode koji vrši predobradu slike. S obzirom da slika ima dvije dimenzije, potrebno je proširiti ju na tri dimenzije kako bi bila kompatibilna sa *U-Net* arhitekturom. Nakon proširenja dimenzija, slika je normalizirana Min-max i Mean normalizacijom, te je normalizirana ImageNet statistikom za bolje rezultate tokom treniranja. Metode za proširenje dimenzija slike i za Min-max i Mean normalizaciju izvedene su pomoću *PyTorch*-ove *Lambda* klase, dok je ImageNet realiziran unosom njegovih specifičnih srednjih vrijednosti i standardnih devijacija u *Normalize* klasu.

```
expand_3d = transforms.Lambda(lambda x: x.view(x.shape).expand(3, -1, -1))
min_max = transforms.Lambda(lambda x: (x - x.min()) / x.max())
mean_val = transforms.Lambda(lambda x: x - x.mean())
normalize = transforms.Normalize([0.485, 0.456, 0.406], [
    0.229, 0.224, 0.225]) # imagenet normalization

image = expand_3d(image)
image = min_max(image)
image = mean_val(image)
image = normalize(image)
```

Sl. 6.6. predobrada slike

5.4.2. Predobrada maske

Nakon predobrade slike započinje predobrada maske. Proces predobrade maske nešto je drugačiji od procesa predobrade slike. Za predobradu maske korištena je pomoćna funkcija *mask2label*. Slike 6.7. i 6.8. prikazuju kod za preobradu maske unutar *x_y_trasnsforms* metode i pomoćnu *mask2label* funkciju.

```
mask_to_label = transforms.Lambda(lambda x: mask2label(x, data_set_name=data_set_name))
mask = mask_to_label(mask)
```

Sl. 6.7. predobrada slike u *x_y_transforms*

```
def mask2label(mask: np.array, data_set_name: str, show_vals: bool = False):
    mask = mask.squeeze()

    if show_vals:
        print("unique values ", np.unique(mask))

    if data_set_name == "CHAOS":
        LIVER = 0.24705882
        SPLEEN = 0.9882353
        L_KIDNEY = 0.7411765
        R_KIDNEY = 0.49411765
        unique_vals = [SPLEEN, R_KIDNEY, L_KIDNEY, LIVER]

    s = mask.shape
    ones, zeros = np.ones(s), np.zeros(s)
    one_hot_map = [np.where(mask == unique_vals[targ], ones, zeros) for targ in range(len(unique_vals))]
    one_hot_map = np.stack(one_hot_map, axis=0).astype(np.float32)

    return one_hot_map
```

Sl. 6.8. *mask2label* funkcija

U *mask2label* funkciji definirane su vrijednosti piksela za svaki abdominalni organ kojeg je potrebno segmentirati. Vrijednosti piksela ručno su pronađene nakon normalizacije slika. Predobrada maske je proces u kojem se stvara segmentacijska mapa. Segmentacijska mapa niz je 2D slika gdje svaka slika predstavlja organ izdvojen iz originalne maske. Organi su izdvojeni na način da se prethodno znala točna vrijednost piksela za svaki organ kojeg je potrebno segmentirati. Funkcija vraća sastavljenu segmentacijsku mapu. Na svakoj segmentacijskog mapi, segmentirani organ je područje gdje je vrijednost piksela jedan, a sve ostalo je pozadina s vrijednošću nula. Ukoliko određeni organ nije bio prisutan na početnoj maski, svi pikseli njegove segmentacijske

mape će imati vrijednost nula. Nakon navedenih procesa gdje su se obradile slike i maske, metoda *x_y_transforms* vraća *tuple* koji predstavlja obrađenu sliku i masku.

5.5. Dinamičko upravljanje podacima

Za dinamičko upravljanje podacima korištena je *DataLoader* klasa. *DataLoader* omogućava efikasno i dinamičko učitavanje podataka tijekom faze treniranja. Koristi se za grupiranje podataka u serije (engl. *batch*), njihovo miješanje, transformaciju i slične procese koji ubrzavaju proces pripreme podataka. Slika 6.9. prikazuje *init_dataloaders* funkciju koja kreira *DataLoader* instance koje će biti iskorištene u fazama treniranja i validacije modela.

```
def init_dataloaders(batch_size: int) -> dict:

    train_set = ChaosDataset(dataset_type='train')
    val_set = ChaosDataset(dataset_type='valid')

    dataloaders = {
        'train': DataLoader(train_set, batch_size=batch_size, shuffle=True, num_workers=0),
        'val': DataLoader(val_set, batch_size=batch_size, shuffle=True, num_workers=0)
    }

    return dataloaders
```

Sl. 6.9. *init_dataloaders* funkcija

Funkcija prvo kreira dva skupa podataka koristeći *ChaosDataset* klasu. Jedna instanca predstavlja skup za treniranje, a druga instanca skup za validaciju. Zatim se kreiraju *DataLoader* instance za navedena dva skupa podataka, gdje se postavljaju parametri kao veličina serije, miješanje podataka i broj radnika (engl. *workers*) za paralelno učitavanje. Instance se vraćaju u obliku *dictionary*-a s ključevima (engl. *key*) „train“ i „val“ koji sadrže odgovarajuće instance *DataLoader* klase za trening i validaciju.

5.6. Funkcija gubitka

Funkcija gubitka igra ključnu ulogu u procesu treniranja modela jer pruža povratne informacije o koje su potrebne kako bi model prilagodio svoje težine (engl. *weights*). Služi kao metrika koja opisuje razliku između predviđenih vrijednosti modela i stvarnih ciljnih vrijednosti. U ovom radu, realizirana je kombinacijom BCE (engl. *binary cross entropy*) metrike i DICE koeficijenta. Funkcija kao parametre prima predviđanje modela, masku, *dictionary* za praćenje rezultata i zadani parametar *bce_weight*. Predviđanje se pretvara u vjerojatnosti pomoću *sigmoid* funkcije, kalkiliraju se obje funkcije gubitka na temelju predviđanja modela i maske te se računa sveukupni gubitak kombinacijom obje metrike, gdje parametar *bce_weight* predstavlja utjecaj BCE metrike na kalkulaciju sveukupnog gubitka. Pojedinačni rezultati i sveukupni rezultat spremaju se u *metrics dictionary*. Funkcija vraća vrijednost sveukupnog gubitka. Formula 6-1 prikazuje odnos BCE-a i DICE koeficijent metrike, dok slika 6.10. prikazuje kod *calc_loss* funkcije.

$$\text{loss} = \text{BCE} * \text{bce}_{\text{weight}} + \text{DICE} * (1 - \text{bce}_{\text{weight}}) \quad (6-1)$$

```
def calc_loss(pred, target, metrics, bce_weight=0.5):
    bce = F.binary_cross_entropy_with_logits(pred, target)
    pred = torch.sigmoid(pred)
    dice = dice_loss(pred, target)
    loss = bce * bce_weight + dice * (1 - bce_weight)

    metrics['bce'] += bce.data.cpu().numpy() * target.size(0)
    metrics['dice'] += dice.data.cpu().numpy() * target.size(0)
    metrics['loss'] += loss.data.cpu().numpy() * target.size(0)

    return loss
```

Sl. 6.10. *calc_loss* funkcija

5.6.1. Binary Cross Entropy

Binary Cross Entropy je funkcija gubitka koja se koristi za binarne klasifikacijske zadatke mjereći razliku između predviđenih i stvarnih binarnih oznaka. Posebno je korisna za zadatke gdje je potrebno predvidjeti prisutnost ili odsutnost određenih klasa, kao što je u ovom slučaju prisutnost određenog abdominalnog organa. Formula 6-2 predstavlja BCE [27].

$$BCE = -\frac{1}{N} \sum_{i=1}^N y_i * \log(p(y_i)) + (1 - y_i) * \log(1 - p(y_i)) \quad (6-2)$$

, gdje je:

- $p(y_i)$ - vjerojatnost vrijednosti 1
- $1 - p(y_i)$ - vjerojatnost vrijednosti 0

5.6.2. DICE koeficijent

DICE(Dice-Sørensen) koeficijent je metrika koja se koristi za procjenu kvalitete segmentacije. Izračunava se kao dvostruka veličina preklapanja između predviđene i stvarne maske, podijeljena s ukupnim brojem piksela u obje maske. Što je veća vrijednost DICE koeficijenta, to znači da je veće preklapanje piksela između predviđene i stvarne maske. Formula 6-3 predstavlja DICE koeficijent [28].

$$DICE(X, Y) = \frac{2 * |X \cap Y|}{|X| + |Y|}$$

, gdje je:

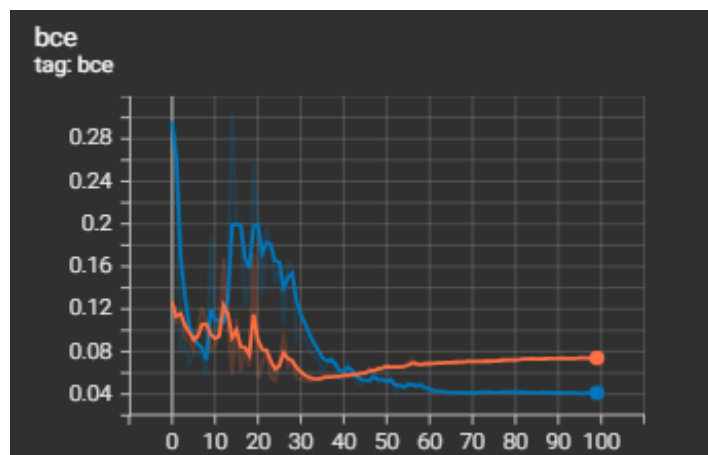
- X - predviđena maska
- Y - stvarna maska

(6-3)

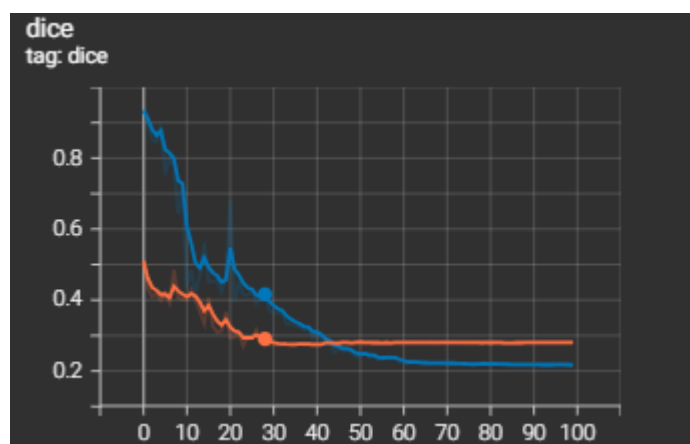
DICE se koristi kao funkcija gubitka tako da se vrijednost koeficijenta oduzme od broja 1.

6. Dobiveni rezultati

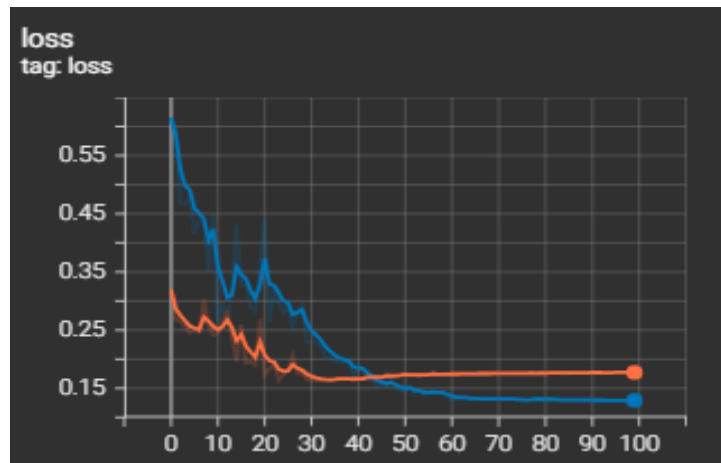
Cilj semantičke segmentacije abdominalnih organa je označavanje svakog piksela unutar slike sa njegovom odgovarajućom klasom, tj. organom koji ga predstavlja. Može se reći da je proces semantičke segmentacije zapravo klasifikacija slike na razini piksela. Za fazu treniranja modela korišteno je 1270 parova MRI slika i maski. Prilikom razdvajanja skupa podataka na skup za treniranje i skup za validaciju, izdvojen je 381 par za validaciju, odnosno trideset posto sveukupnog skupa. Skup za treniranje sastoji se od 889 parova. Dimenzije slika i maski bile su 96x96 piksela. Model je treniran kroz 100 epoha, pri čemu je veličina serije bila 32 para. Kao optimizator korišten je Adam sa stopom učenja od 0,001. Oba modela, ResNet U-Net i U-Net++, trenirana su na istom skupu podataka. Slike 7.1., 7.2. i 7.3. prikazuju krivulje funkcija gubitka tokom faze treniranja za oba modela, gdje narančasta krivulja predstavlja krivulju gubitka na ResNet U-Net-u, a plava na U-Net++.



Sl. 7.1. krivulje gubitka prema BCE



Sl. 7.2. krivulje gubitka prema DICE koeficijentu



Sl. 7.3. krivulje sveukupnog gubitka

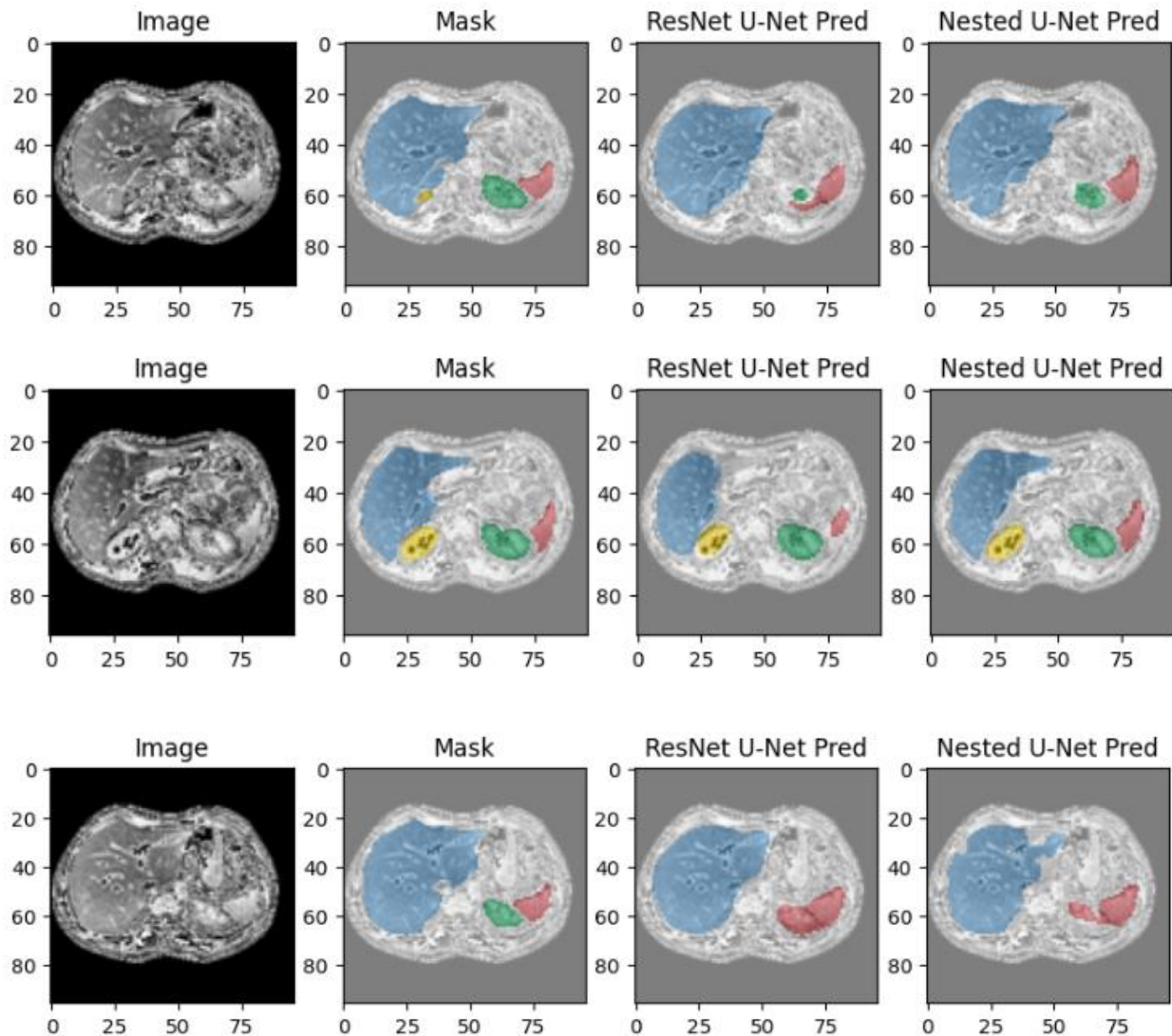
Nakon analize prethodnih grafova i generiranih datoteka, može se zaključiti:

- U-Net++ imao je bolje rezultate na validaciji
- ResNet U-Net je bio brži u treniranju i zauzima manje memorije od U-Net++ (ResNet U-Net oko 70 MB, U-Net++ oko 560 MB)
- Pri početnoj fazi treniranja, vidljivo je da ResNet U-Net ima bolje performanse, vjerojatno zbog manjeg broja parametara i/ili zbog iskorištenih težina ResNet-a

Prilikom treniranja oba modela, praćene su najbolje vrijednosti gubitaka. Najbolja vrijednost gubitaka na ResNet U-Net-u je 0,163, a na U-Net++ 0,129. Ovo ukazuje da je U-Net++ model postigao bolji učinak u smislu smanjenja gubitka tokom treniranja. Niža vrijednost gubitka na U-Net++ modelu može biti rezultat bolje sposobnosti ovog modela da uči i generalizira podatke, što se može pripisati njegovoj složenijoj arhitekturi i mogućnosti obrade složenijih uzoraka u podacima.

6.1. Predviđanja modela na validacijskom skupu

Slike 7.4. prikazuju predviđanja oba modela uz stvarnu sliku i stvarnu masku. Predviđanja su bila nad validacijskim skupom. Jetra je označena plavom bojom, slezena crvenom, desni bubreg zelenom, a lijevi bubreg žutom bojom.

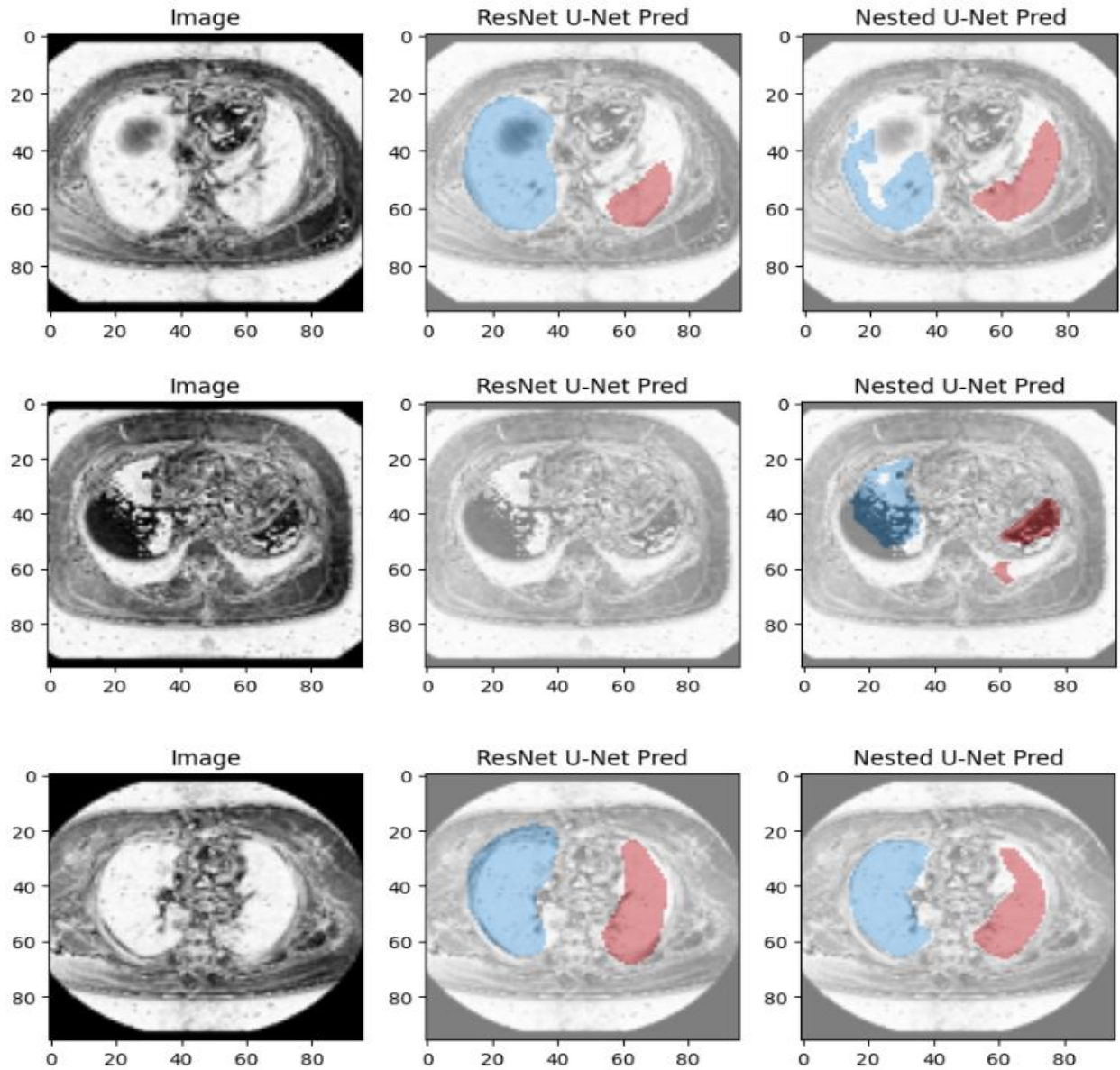


Sl. 7.4. Segmentacija organa na validacijskom skupu

Primjetno je kako oba modela uspješno segmentiraju organe. Prostim okom vidljivo je kako U-Net++ ipak ima nešto malo točniju segmentaciju organa od ResNet U-Net-a.

6.2. Predviđanja modela na testnom skupu

Slike 7.5. prikazuju predviđanja oba modela na testnom skupu, tj. nad modelu neviđenim podacima. Boje koje označuju organe jednake su kao i u prošlom odlomku.



Sl. 7.5. Segmentacija organa na testnom skupu

7. Zaključak

U radu su predstavljene metode za segmentaciju četiri abdominalna organa iz MRI medicinskih snimki koristeći konvolucijske neuronske mreže temeljene na U-Net arhitekturi. Cilj je bila segmentacija jetre, slezene, te lijevog i desnog bubrega. Praktični dio rada izveden je pomoću programskog jezika Python i razvojnog okvira PyTorch uz pomoćne biblioteke poput NumPy i Tensorboard-a. Korišten je skup podataka CHAOS koji je sadržavao MRI medicinske slike u DICOM formatu i njihove odgovarajuće maske koje su bile u *.png* formatu. Istrenirana i vrednovana su dva modela temeljena na U-Net arhitekturi: ResNet U-Net i U-Net++. Oba modela uspješno su segmentirala tražene abdominalne organe, gdje je U-Net++ imao nešto bolje rezultate. Mjesta za napredak još uvijek ima. Performanse modela bi se mogle poboljšati procesima poput proširenja skupa podataka, augmentacijom podataka, te dodatnom modifikacijom modela.

Literatura

- [1] G. K., Murugesan *i ostali*, „Automatic Abdominal Multi Organ Segmentation using Residual UNet“, Bioengineering, preprint, velj. 2023.
- [2] „Meet nnUNet — paddle-nnUNet 1.0.0 documentation“ [online]. Dostupno na: https://paddle-nnnet.readthedocs.io/zh_CN/latest/introduce_nnnet.html. [Pristupljeno: 13.7.2023.].
- [3] S., Kondo, S., Kasai, „Multi-Modality Abdominal Multi-Organ Segmentation with Deep Supervised 3D Segmentation Model“.
- [4] „The architecture of 3D Res U-Net“ [online]. Dostupno na: https://www.researchgate.net/figure/The-architecture-of-3D-Res-U-Net_fig3_356015151. [Pristupljeno: 13.7.2023.].
- [5] Y., Wang, Y., Zhou, W., Shen, S., Park, E. K., Fishman, A. L., Yuille, „Abdominal multi-organ segmentation with organ-attention networks and statistical fusion“, *Med. Image Anal.*, sv. 55, str. 88–102, srp. 2019.
- [6] „Slezena“, *Wikipedija*. 21-pros-2021.
- [7] „Biology-5-Homeostasis-and-response_Knowledge-MAT-Part-2_GCSE.pdf“.
- [8] „Bubreg“, *Wikipedija*. 02-sij-2022.
- [9] „Anatomy and Function of the Liver“ [online]. Dostupno na: <https://www.stanfordchildrens.org/en/topic/default?id=anatomy-and-function-of-the-liver-90-P03069>. [Pristupljeno: 14.7.2023.].
- [10] „Hrvatska: KB Dubrava dobila vrijedan CT uređaj za tretman pacijenata sa koronavirusom“ [online]. Dostupno na: <https://www.aa.com.tr/ba/balkan/hrvatska-kb-dubrava-dobila-vrijedan-ct-uredaj-za-tretman-pacijenata-sa-koronavirusom-/1822866>. [Pristupljeno: 14.7.2023.].
- [11] FDA, *English: Drawing of CT fan beam and patient in a CT imaging system*. 2019.
- [12] „Magnetska rezonanca - Sv. Katarina, Zagreb“ [online]. Dostupno na: <https://www.svkatarina.hr/radiologija-i-dijagnostika/magnetska-rezonanca>. [Pristupljeno: 14.7.2023.].
- [13] „Magnetic resonance imaging (MRI) (article)“ [online]. Dostupno na: <https://www.khanacademy.org/test-prep/mcat/physical-processes/proton-nuclear-magnetic-resonance/a/magnetic-resonance-imaging-mri>. [Pristupljeno: 14.7.2023.].
- [14] N., Bhargav, „Neurons in Neural Networks | Baeldung on Computer Science“ [online], 25-kol-2022. Dostupno na: <https://www.baeldung.com/cs/neural-networks-neurons>. [Pristupljeno: 14.7.2023.].
- [15] „Introduction to Convolutional Neural Networks CNNs“ [online]. Dostupno na: <https://aigents.co/data-science-blog/publication/introduction-to-convolutional-neural-networks-cnns>. [Pristupljeno: 14.7.2023.].
- [16] „Figure 10. A comparison of max-pooling with average-pooling for...“ [online]. Dostupno na: https://www.researchgate.net/figure/A-comparison-of-max-pooling-with-average-pooling-for-aggregation-across-activation_fig5_327741382. [Pristupljeno: 14.7.2023.].
- [17] „Figure 3. The difference between standard NN and Dropout NN [53]...“ [online]. Dostupno na: https://www.researchgate.net/figure/The-difference-between-standard-NN-and-Dropout-NN-53-Dropout-method-randomly-drops_fig3_336754568. [Pristupljeno: 14.7.2023.].
- [18] „Why do we use Batch Normalization in Deep Learning and How do we optimize Network ?“ [online]. Dostupno na: <https://www.linkedin.com/pulse/why-do-we-use-batch-normalization-deep-learning-how-optimize-shah>. [Pristupljeno: 14.7.2023.].
- [19] F., Kulić, „KLASIFIKACIJA OBJEKATA DETEKTIRANIH ISPRED VOZILA POMOĆU KAMERE NA PREDNJOJ STRANI VOZILA“.

- [20] „Gambar 2.12 Arsitektur Convolutional Neural Network“ [online]. Dostupno na: https://www.researchgate.net/figure/Gambar-212-Arsitektur-Convolutional-Neural-Network_fig1_346444515. [Pristupljeno: 14.7.2023.].
- [21] „PyTorch“, *Wikipedia*. 10-svi-2024.
- [22] M., Ali, „U-NET Architecture Explained and Implementation“ [online], 21-svi-2022. Dostupno na: <https://becominghuman.ai/u-net-architecture-explained-and-implementation-470a5095ad57>. [Pristupljeno: 18.8.2023.].
- [23] N., Singla, „UNet with ResBlock for Semantic Segmentation“ [online], 12-pros-2019. .
- [24] S.-H., Tsang, „Review: UNet++ — A Nested U-Net Architecture (Biomedical Image Segmentation)“ [online], 01-lis-2019. .
- [25] A. E., Kavur, M. A., Selver, O., Dicle, M., Barış, N. S., Gezer, „CHAOS - Combined (CT-MR) Healthy Abdominal Organ Segmentation Challenge Data“. Zenodo, 11-tra-2019.
- [26] „torchvision.transforms — Torchvision master documentation“ [online]. Dostupno na: <https://pytorch.org/vision/0.9/transforms.html>. [Pristupljeno: 17.5.2024.].
- [27] D., Godoy, „Understanding binary cross-entropy / log loss: a visual explanation“ [online], 10-srp-2022. Dostupno na: <https://towardsdatascience.com/understanding-binary-cross-entropy-log-loss-a-visual-explanation-a3ac6025181a>. [Pristupljeno: 23.5.2024.].
- [28] „Dice-Sørensen coefficient“, *Wikipedia*. 19-svi-2024.

Duboko učenje za segmentaciju različitih abdominalnih organa iz medicinskih slika

Sažetak

Segmentacija abdominalnih organa korištenjem neuronskih mreža predstavlja jedan od ključnih koraka u modernoj medicinskoj dijagnostici i planiranju tretmana. Korištenjem MRI snimki za treniranje modela, konvolucijske neuronske mreže mogu precizno identificirati i segmentirati različite organe u abdominalnoj šupljini. Ovi modeli su u stanju naučiti složene obrasce i karakteristike koje omogućuju razlikovanje između organa kao što su jetra, slezena i bubrezi. Automatska segmentacija smanjuje potrebu za ručnim označavanjem od strane radiologa, poboljšava točnost dijagnoza i ubrzava proces analize medicinskih snimaka. U ovom radu predstavljene su dvije metode segmentacije abdominalnih organa koristeći dva modela temeljena na U-Net arhitekturi neuronskih mreža. Modeli su trenirani, vrednovani, testirani i uspoređeni. Oba modela uspješno su odrađivala segmentaciju, svaki sa svojom razinom preciznosti. Preciznosti oba modela bi se mogle još dodatno poboljšati proširenjem skupa podataka.

Ključne riječi:

abdominalni organi, duboko učenje, konvolucijske neuronske mreže, magnetska rezonancija, semantička segmentacija

Deep learning for segmentation of different abdominal organs from medical images

Abstract

Segmentation of abdominal organs using neural networks represents one of the key steps in modern medical diagnostics and treatment planning. Using MRI images for model training, convolutional neural networks can accurately identify and segment different organs in the abdominal cavity. These models can learn complex patterns and features that enable the differentiation between organs such as the liver, spleen, and kidneys. Automatic segmentation reduces the need for manual annotation by radiologists, improves diagnostic accuracy, and speeds up the process of analyzing medical images. This paper presents two methods for segmenting abdominal organs using two models based on the U-Net neural network architecture. The models were trained, validated, tested and compared with both models successfully performing segmentation, each with its own level of precision. The precision of both models could be further improved by expanding the dataset.

Keywords:

abdominal organs, deep learning, convolutional neural network, magnetic resonance imaging, semantic segmentation