

Geometrijsko duboko učenje za segmentaciju urođenih srčanih mana iz CT slika

Kvesić, David

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:200:607463>

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

Download date / Datum preuzimanja: **2025-04-03**

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 diplomski studij
Računarstvo**

**GEOMETRIJSKO DUBOKO UČENJE ZA
SEGMENTACIJU UROĐENIH SRČANIH MANA IZ CT
SLIKA**

Diplomski rad

David Kvesić

Osijek, 2024.

**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:	David Kvesić
Studij, smjer:	Sveučilišni diplomski studij Računarstvo
Mat. br. pristupnika, god.	D-1217R, 06.10.2021.
JMBAG:	0165078857
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:	Geometrijsko duboko učenje za segmentaciju urođenih srčanih mana iz CT slika
Znanstvena grana diplomskog rada:	Umjetna inteligencija (zn. polje računarstvo)
Zadatak diplomskog rada:	Istražiti i opisati način dobivanja medicinskih CT slika i njihove najvažnije karakteristike. Istražiti i opisati kliničku pozadinu (dijelove srca, srca sa 16 osnovih tipova urođenih srčanih mana te problematiku obrade takvih slika). Dati kratak pregled područja prethodnih istraživanja. Objasniti teorijske osnove geometrijskog dubokog učenja s naglaskom na neuronske mreže temeljene na teoriji grafova. Razviti jedinstveni sustav za segmentaciju CT slika sa urođenim srčanim manama pomoću geometrijskih neuronskih mreža. Prikazati i objasniti rezultate te odrediti preciznost
Datum ocjene pismenog dijela diplomskog rada od strane mentora:	23.09.2024.
Ocjena pismenog dijela diplomskog rada od strane mentora:	Izvrstan (5)
Datum obrane diplomskog rada:	1. 10. 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:	02.10.2024.

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK****IZJAVA O IZVORNOSTI RADA**

Osijek, 02.10.2024.

Ime i prezime Pristupnika:

David Kvesić

Studij:

Sveučilišni diplomski studij Računarstvo

Mat. br. Pristupnika, godina upisa:

D-1217R, 06.10.2021.

Turnitin podudaranje [%]:

11

Ovom izjavom izjavljujem da je rad pod nazivom: **Geometrijsko duboko učenje za segmentaciju urođenih srčanih mana iz CT 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	1
2. PRETHODNA RJEŠENJA ZA SEGMENTACIJU SLIKA	3
3. MEDICINSKA POZADINA SRCA	8
3.1. Urođene srčane mane	10
4. OBRADA MEDICINSKIH SLIKA	13
4.1. Zapis i obrada medicinskih slika	14
4.1.1. NIFTI	15
4.2. Teorijske osnove geometrijskog dubokog učenja	15
4.2.1. Neuronske mreže	15
4.2.2. Konvolucijske neuronske mreže	16
4.2.3. Generative Adversarial Network (GAN)	18
5. SUSTAV ZA SEGMENTACIJU CT SLIKA	21
5.1. Razvijeni sustav	21
5.2. Rezultati	30
6. ZAKLJUČAK	33
LITERATURA	34
SAŽETAK	37
ABSTRACT	38

1. UVOD

Kardiovaskularne bolesti vodeći su uzrok smrti danas u svijetu i u konstantnom su porastu. Glavni su uzroci takvog porasta smanjena fizička aktivnost i lošija prehrana koji su uzrokovani ubrzanim načinom života. Bitan dio tog problema su urođene srčane mane koje, ako nisu otkrivene na vrijeme, mogu prouzročiti ozbiljne zdravstvene probleme. Napretkom tehnologije u medicini došlo je do snažnog razvoja obrade medicinskih slika. Obrada slika je interdisciplinarna istraživačka grana koja obuhvaća medicinu, računarstvo, matematiku, inženjerstvo i druga područja. Segmentacija slike je proces dijeljenja slike na više segmenata ili regija, koje predstavljaju značajne dijelove slike, kao što su objekti ili granice između objekata. Cilj segmentacije je pojednostaviti ili promijeniti prikaz slike kako bi se olakšala analiza, prepoznavanje ili razumijevanje. Na primjer, u medicinskim slikama segmentacija može pomoći u identifikaciji različitih tkiva, lezija ili organa poput srca i srčanih komora.

Segmentacija se primjenjuje na snimkama poput računalne tomografije (CT) i magnetske rezonancije (MRI), te predstavlja jedan od najzahtjevnijih zadataka u obradi medicinskih slika. Metode segmentacije mogu uključivati tehnike kao što su klasteriranje piksela, otkrivanje rubova i napredne metode poput konvolucionih neuronskih mreža (CNN).

Rad je organiziran u pet poglavlja. U prvom poglavlju dan je pregled nekih postojećih rješenja koja se koriste za segmentaciju slika. U drugom poglavlju opisana je medicinska pozadina srca i dan je kratki pregled urođenih srčanih mana s njihovim opisom i klasifikacijom. U trećem poglavlju opisan je proces obrade medicinskih slika i teorijske osnove dubokog učenja potrebne za implementaciju rješenja. U četvrtom poglavlju detaljnije je opisano programsko rješenje implementirano za zadatak segmentacije CT slika. Peto i posljednje poglavlje je zaključak nakon provedenog rada.

1.1. Zadatak diplomskog rada

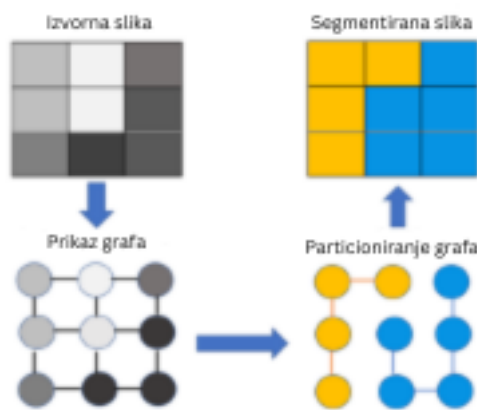
Zadatak diplomskog rada je istražiti i opisati način dobivanja medicinskih CT slika i njihove najvažnije karakteristike te opisati kliničku pozadinu urođenih srčanih mana. Dati kratak pregled područja prethodnih istraživanja i objasniti teorijske osnove geometrijskog dubokog učenja s naglaskom na neuronske mreže. Razviti jedinstveni sustav za segmentaciju CT slika sa urođenim

srčanim manama pomoću geometrijskih neuronskih mreža i analizirati rezultate dobivene izvođenjem razvijenog sustava.

2. PRETHODNA RJEŠENJA ZA SEGMENTACIJU SLIKA

Segmentacija CT slika srca bavi se dekompozicijom slike u njene sastavne dijelove i ima ključnu ulogu u dekompoziciji slika na anatomske značajne regije, poznate i kao područja interesa. Tipične regije na koje se segmentacija fokusira su četiri komore u srcu: lijeva i desna pretklijetka te lijeva i desna klijetka ili cijela srčana anatomija uključujući četiri komore, koronarne arterije i abdominalnu aortu. Budući da je ručni proces srčane segmentacije iznimno naporan i dugotrajan za radiologe kreirane su polu automatske i potpuno automatizirane metode segmentacije [1].

Segmentacija temeljena na grafovima (eng. Graph-Based Segmentation, (GBS)) je metoda segmentacije koja primjenjuje principe teorije grafova kako bi se slika rekonstruirala u obliku grafa. Svaki piksel predstavlja čvor dok su bridovi stupanj sličnosti između susjednih čvorova. Sljedeći je korak razdvajanje grafa u regije koristeći neki od algoritama za particioniranje grafova (Slika 2.1). Cilj algoritma je minimizirati vrijednosti bridova između čvorova. Najpoznatiji algoritmi za particioniranje grafova su Kernighan-Lin algoritam, K-way algoritam i spektralni klastering. GBS je precizna i fleksibilna metoda segmentacije slika, ali zahtijeva vrlo oprezno postavljanje parametara te veliko korištenje vremena i računalnih resursa [2].



Slika 2.1 Prikaz transformacije iz originalne u segmentiranu sliku [2]

Thresholding je metoda kojom se odvajaju pikseli čija je vrijednost, odnosno intenzitet veći od praga pri čemu se slika dijeli na prednju stranu i pozadinu. Navedenom tehnikom se iz slike sa spektrom boja dobiva binarna slika u kojoj su crnom bojom označeni pikseli s većom vrijednosti

od praga, a bijelom bojom oni koji su manji. Za razliku od konvencionalnog (globalnog) tresholdinga gdje je vrijednost praga konstantna, kod adaptivnog tresholdinga vrijednost praga se dinamički mijenja prelaskom preko slike. Ova metoda pokazala se osobito korisnom kod obrade slika s raznim pozadinskim osvjetljenjima. Najpoznatije tehnike globalnog tresholdinga su Otsu metoda i Kapurova metoda [3].

Otsu metoda razdvaja sliku na dva dijela, pozadinu i prednji dio. Zatim koristi grayscale histogram kako bi pronašao optimalni prag po kojem će razdvojiti dvije regije. Histogram je definiran jednadžbom 2-1:

$$H(i) = \sum_{x=1}^M \sum_{y=1}^N [I(x, y) = i] \quad (2-1)$$

gdje su M i N širina i duljina slike, a $[I(x, y) = i]$ funkcija koja vraća 1 ako je piksel na poziciji (x, y) veći od praga, a 0 ako nije. Prednosti Otsu metode su jednostavnost i brzina kao i to da ne zahtijeva predznanje o procesiranoj slici te može automatski odrediti prag za razdvajanje slike od pozadine. Glavni nedostatak ove metode je što može odrediti samo jednu vrijednost praga zbog čega daje lošije rezultate ako histogram ima više od jednog vrha. Iz istog razloga metoda je osjetljiva i na nejednako osvjetljenje na slici kao i na prisutnost šumova što su sve faktori koji utječu na nepreciznost vrijednosti praga[4].

Kapurova metoda popularna je metoda za određivanje više vrijednosti praga. Metoda koristi maksimiziranje entropije histograma, poznato i kao Kapurova entropija, kako bi dobila optimalne vrijednosti tresholda. Ako su $[th_1, th_2, \dots, th_n]$ kombinacije tresholda koje dijele sliku u različite kategorije onda je Kapurova metoda definirana kao 2-2:

$$H(th_1, th_2, \dots, th_n) = H_0 + H_1 + \dots + H_n \quad (2-2)$$

gdje je:

$$H_n = - \sum_{j=th_n}^{L-1} \frac{p_j}{\omega_n} \ln \frac{p_j}{\omega_n}, \omega_n = \sum_{j=th_n}^{L-1} p_j \quad (2-3)$$

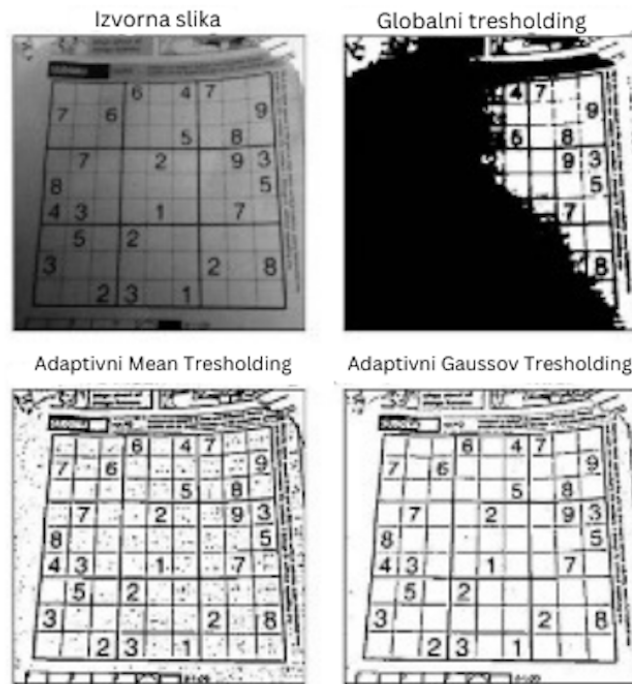
H_0, H_1, \dots, H_n su entropije određenih klasa, $\omega_0, \omega_1, \dots, \omega_n$ su njihove vjerojatnosti. Kako bi se održale optimalne vrijednosti praga koristi se maksimizacija pomoću formule 2-4:

$$f_{Kapur}(th_1, th_2, \dots, th_n) = arg arg H (th_1, th_2, \dots, th_n) \quad (2-4)$$

Problem kod povećanja broja praga je da kompleksnost raste eksponencijalno, zbog čega Kapurova metoda nije efikasna za računanje većeg broja praga [5].

Nedostatke globalnog thresholdinga nadomještuje adaptivni thresholding koji za razliku od globalnog, računa vrijednosti thresholda lokalno uzimajući u obzir svaku regiju zasebno i omogućava očuvanje detalja na slici jer prilagođava vrijednosti bazirane na lokalnim intenzitetima piksela. Najpoznatija metoda adaptivnog thresholdinga je Gaussov adaptivni thresholding (eng. *Gaussian Adaptive Thresholding*)

Na slici 2.2 prikazana je usporedba globalnog i adaptivnog thresholdinga. Iz slike je vidljivo poboljšanje slike kao rezultat odabira različitog praga na određenim područjima slike. U ovom slučaju veliku ulogu igra sjena, odnosno osvjetljenje zbog kojeg globalni thresholding ne daje zadovoljavajući rezultat. Na donjim slikama uspoređeni su i rezultati dvaju pristupa adaptivnom thresholdingu. Adaptive Mean Thresholding uzima prosječnu vrijednost piksela u svakoj regiji koju računa, dok Adaptive Gaussian Thresholding koristi Gaussovu težinsku raspodjelu, dajući prednost pikselima u središtu regije. Rezultat je još čišća obrada slike.



Slika 2.2 Prikaz adaptivnog tresholdinga [6]

S razvojem dubokog učenja, razvijene su napredne metode za segmentaciju medicinskih slika. U radu [7] predstavljena je segmentacija CT snimaka srca korištenjem K-Means klasteriranja i metode matematičke morfologije. Budući da je ručno označavanje segmenata vrlo zahtjevno, primijenjeno je nenadzirano učenje, odnosno pristup bez korištenja unaprijed označenih podataka za treniranje modela. K-Means algoritam grupira podatke u k klastera na temelju njihove udaljenosti od središta klastera. Nakon inicijalne dodjele podataka, središta klastera se ažuriraju na temelju novih podataka, a taj proces se ponavlja iterativno dok ne budu ispunjeni zadani kriteriji konvergencije.

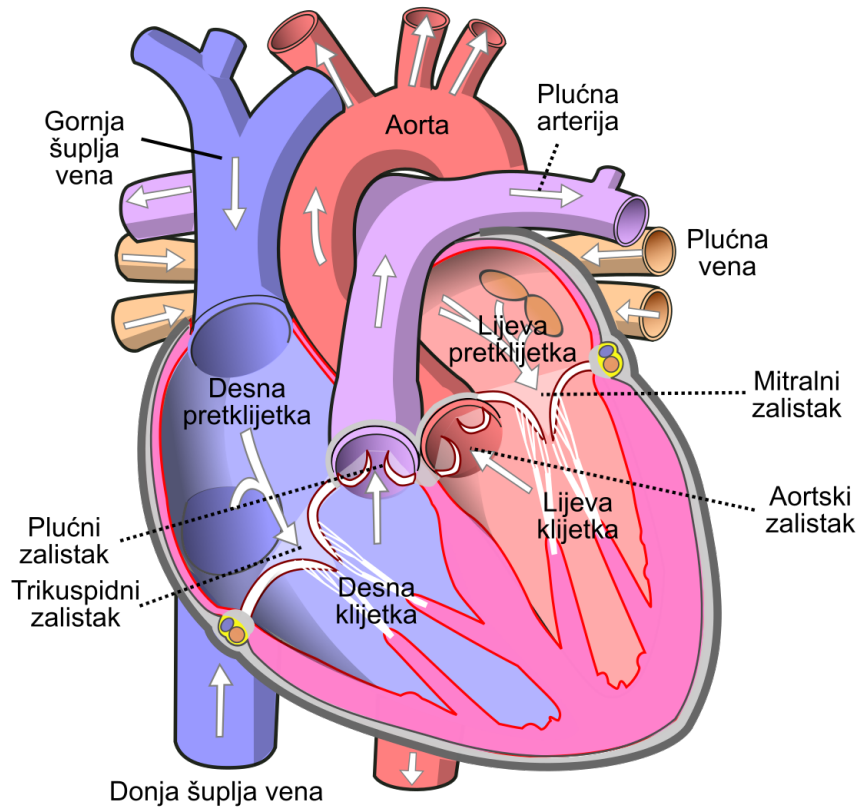
Eksperiment je proveden na uzorku od 500 subjekata u dva scenarija. U prvom slučaju, slike su sadržavale cijeli volumen srca, pri čemu je svaka snimka imala 56 slika. U drugom scenariju, slike su obuhvaćale samo gornju polovicu srca prije pojave jetre, sa 30 slika po snimci. Prosječni rezultat K-Means metode iznosio je 0,4130. Kod slika koje su prikazivale cjelokupan volumen srca, postignuta je ukupna točnost od 34,90% uz mIoU (mean Intersection over Union) vrijednost od 41,26%. Snimke s gornjom polovicom srca dale su bolje rezultate, s točnošću od 55,10% i mIoU od 71,46%. Lošiji rezultati u prvom scenariju pripisuju se neuspješnom filtriranju jetre, kao i

poteškoćama u generiranju maske za kralježnicu, što je negativno utjecalo na konačne rezultate segmentacije.

U radu [8] provedena je segmentacija dijelova srca iz 4D CT slika pojačanih kontrastnim sredstvom, pomoću modela dubokog učenja koristeći konvolucijske neuronske mreže. Automatska segmentacija srčane komore i miokarda lijeve klijetke tijekom srčanom ciklusa značajno proširuje upotrebu CT-a srca s kontrastnim sredstvom, što potencijalno omogućuje dubinsku procjenu funkcije srca. U ovom istraživanju korištene su 4D kontrastno pojačane CT snimke srca pacijenata koji su izabrani za trans-katetersku implantaciju aortnog zaliska. Korišteno je 21605 3D slika od 1509 pacijenata od kojih su slike 12 pacijenata odabrane za razvojni dio, a slike preostalih 1497 za skup testnih podataka. Trodimenzionalne konvolucijske neuronske mreže trenirane su sa sistoličkim i dijastoličkim snimkama. Segmentacija je rezultirala s Dice koeficijentom od $0,89 \pm 0,10$ i srednjom simetričnom površinskom udaljenosti (ASSD) od $1,43 \pm 1,45\text{mm}$ za 12 pacijenata u 3D-u i Dice koeficijentom $0,89 \pm 0,08$ i ASSD $1,86 \pm 1,20\text{mm}$ za 81 pacijenta u 2D-u. Kvalitativna evaluacija cijelog testnog skupa podataka od 1497 pacijenata pokazala je da je segmentacija ocijenjena ocjenom 1 (klinički uspješna) u 98,5% slučajeva za šupljinu lijeve klijetke, 92,2% za miokard, 83,1% za lijevu pretklijetku, 96,3% za lijevu klijetku te 91,6% za desnu klijetku. Zaključeno je kako je automatska metoda koristeći konvolucijske neuronske mreže kroz srčani ciklus na velikom skupu 4-dimenzionalnih srčanih CT slika postigla klinički uspješan rezultat, koji potencijalno omogućuje dubinsku procjenu srčanih funkcija.

3. MEDICINSKA POZADINA SRCA

Ljudsko srce mišićni je organ veličine stisnute ljudske šake koji ima dvije glavne funkcije: prikupljanje kisikom osiromašene krvi iz cijeloga tijela i pumpanje iste krvi u pluća te prikupljanje kisikom obogaćene krvi iz pluća i pumpanje krvi u sva tkiva u tijelu. Srce se nalazi u središtu prsa, zaštićeno iza prsne kosti, između plućnih krila, pozicionirano iznad dijafragme. Srčana stijenka omotana je u perikard, a sastavljena je od tri sloja: epikarda, miokarda i endokarda. Perikard je zaštitna vrećica oko srca sa dva sloja vezivnog tkiva i tekućinom koja sprječava kontakt s drugim dijelovima tijela i olakšava kretanje srca. Epikard je vanjski sloj srčane stijenke, često zvan i visceralni perikard i ima zaštitnu ulogu srca jer olakšava kretanje srca u srčanoj vrećici. Miokard je središnji, mišićni sloj, sastavljen od srčanog mišićnog tkiva zadužen za kontrakciju i relaksaciju srca. Endokard je unutarnji sloj srčane stijenke koji omogućava efikasan protok krvi kroz srce [9]. Unutarnji dio srca sastoji se od četiri šupljine. Gornje dvije šupljine zvane pretkljetke ili atrijske služe za prikupljanje krvi iz tijela, a donje dvije šupljine zvane kljetke ili ventrikuli služe za pumpanje krvi iz srca. Lijevu kljetku i pretkljetku od desnih odvaja interatrioventrikularni septum i dijeli srce na dvije anatomski i fiziološki zasebne jedinice, često zvane lijevo i desno srce. Kljetke i pretkljetke odvojene su srčanim zaliscima, trikuspidalnim u desnom srcu i mitralnim u lijevom srcu [9]. Prikaz glavnih dijelova srca prikazan je na slici 3.1.



Slika 3.1 Prikaz glavnih dijelova ljudskog srca [10]

Ljudsko srce središte je kardiovaskularnog sustava, čija je funkcija opskrba cijelog tijela kisikom i hranjivim tvarima. Kardiovaskularni sustav sastavljen je od dva krvotoka: malog ili plućnog i velikog ili sustavnog krvotoka. Krvotok započinje između dva otkucaja srca kada krv odlazi iz pretkljetki u kljetke čime se one šire. Sljedeća faza naziva se period izbacivanja, kada lijeva i desna kljetka pumpaju krv u arterije. U velikom krvotoku lijeva kljetka pumpa krv obogaćenu kisikom u glavnu arteriju, odnosno aortu. Krv zatim teče iz aorte u manje arterije i mrežu kapilara. Tamo krv predaje kisik, a prikuplja ugljikov dioksid. Tada krv osiromašena kisikom putuje u vene i odlazi u desnu pretkljetku kroz gornju šuplju venu. Taj dio procesa krvotoka naziva se veliki krvotok. Nakon velikog slijedi proces malog krvotoka kada krv osiromašena kisikom iz desne kljetke odlazi putem plućne arterije preko manjih arterija i kapilara u pluća. Tamo se u plućnim kapilarnim mrežama pored plućnih alveola u procesu disanja ugljikov dioksid ispušta iz krvi, a kisik dobiven udisanjem ispušta u krv i tako ulazi u krvotok. Izdisanjem se ugljikov dioksid izbacuje iz tijela, a svježe oksigenirana krv putem plućne vene odlazi u lijevu pretkljetku [11].

3.1. Urođene srčane mane

Urođene srčane mane najčešći su tip urođenih defekata kod ljudi. Prisutni su pri rođenju djeteta i mogu utjecati na način funkcioniranja srca. Srčane mane mogu utjecati na način na koji krv protječe kroz srce i ostatak tijela, a variraju od blagih poput male rupice u srcu do kritičnih kada dijelovi srca ili nedostaju ili su pogrešno formirani. Otprilike jedno od četiri djeteta rođeno s urođenom srčanom manom ima kritičnu srčanu manu i treba operaciju ili druge procedure unutar prvih dvanaest mjeseci života.

U nastavku su opisani glavni tipovi urođenih srčanih mana, koji se mogu podijeliti na kritične i nekritične [12] [13]. U kritične srčane mane ubrajaju se: disekcija aorte (CoA), desni ventrikul s dvostrukim izlazom, D – transpozicija velikih arterija, Ebsteinova anomalija, Sindrom hipoplastičnog lijevog srca, plućna atrezija, tetralogija Fallot, potpuni anomalni utok plućnih vena (TAPVR), trikuspidalna atrezija i arterijski trunkus. Nekritične srčane mane su: otvoreni foramen ovale (PFO), atrijalni septalni defekt (ASD), atrioventrikularni septalni defekt (AVSD), ventrikularni septalni defekt (VSD) te arterijski duktus (PDA).

Disekcija aorte urođena je mana kod koje je dio aorte uži od uobičajenog. Opasnost od pojave ovisi o tome koliko je suženje. Primjerice, kod većih suženja srce mora raditi jače, a s vremenom dovodi do slabljenja srčanog mišića i potencijalno, srčanog udara.

Desni ventrikul s dvostrukim izlazom (dvostruka desna klijetka) srčana je greška pri kojoj i plućna arterija i aorta izlaze iz desne klijetke. U normalnom srcu plućna arterija šalje krv siromašnu kisikom iz desne klijetke u pluća, dok aorta šalje krv bogatu kisikom u ostatak tijela, a u ovom slučaju obje velike žile spajaju se na desnu klijetku. Komunikacija između lijeve klijetke i aorte moguća je samo u slučaju Ventrikularnog septalnog defekta, odnosno rupe u zidu između dvije komore, koja omogućava da dio oksigenirane krvi iz lijeve klijetke dođe do aorte.[14]

D – transpozicija velikih arterija najčešći je oblik transpozicije arterija kod kojeg su dvije glavne arterije, aorta i plućna arterija, zamijenjene, odnosno u transpoziciji. Posljedica toga je da lijeva klijetka pumpa deoksigeniranu krv natrag u tijelo, a desna klijetka pumpa oksigeniranu krv natrag prema plućima.

Ebsteinova anomalija miopatija je desne klijetke s teškom trikuspidalnom regurgitacijom. Trikuspidalni zalistak je formiran niže nego uobičajeno što uzrokuje pumpanje krvi natrag u desnu

pretklijetku umjesto u plućnu arteriju. Ovo stanje može biti popraćeno s atrijalnim septalnim defektom [15].

Sindrom hipoplastičnog lijevog srca srčani je defekt kod kojega je lijeva strana srca nedovoljno razvijena. Uobičajene pojave su nerazvijena lijeva klijetka i mitralni zalistak, a moguća je i pojava atrijalnog septalnog defekta.

Plućna atrezija urođeni je defekt srca kod kojega se zalistak koji kontrolira protok krvi iz srca u plućnu arteriju uopće ne formira te krv ne može teći u plućnu arteriju. Ovaj defekt može biti popraćen i ventrikularnim septalnim defektom. Stanje u kojem je zalistak formiran, ali nedovoljno razvijen naziva se stenoza plućnog zaliska (pulmonarna stenoza).

Tetralogija Fallot je kompleksna srčana anomalija koja se sastoji od četiri defekta: ventrikularnog septalnog defekta (VSD), plućne stenoze, pomaka aortalnog zaliska i hipertrofije desne klijetke. Ventrikularni septalni defekt predstavlja otvor u pregradi između lijeve i desne klijetke, što dovodi do miješanja oksigenirane i deoksigenirane krvi. Plućna stenoza se odnosi na suženje plućnog zaliska i glavne plućne arterije, otežavajući protok krvi iz desne klijetke prema plućima. Aortalni zalistak je proširen i postavljen iznad ventrikularnog septalnog defekta, što uzrokuje dodatno miješanje krvi iz obje klijetke. Zbog povećanog napora desne klijetke uslijed suženja plućne arterije, dolazi do ventrikularne hipertrofije, odnosno zadebljanja mišićnog zida desne klijetke. Ova kombinacija malformacija rezultira smanjenim protokom krvi prema plućima i nižom razinom kisika u organizmu.

Potpuni anomalni utok plućnih vena je stanje u kojem oksigenirana krv odlazi u desnu stranu srca umjesto u lijevu, što dovodi do miješanja arterijske i venske krvi, a posljedica je smanjena zasićenost kisikom djeteta. Dijete obično razvije i atrijalni septalni defekt kako bi oksigenirana krv stigla u lijevo srce, što je nužno za preživljavanje.

Trikuspidalna atrezija je srčana mana kod koje se trikuspidalni zalistak uopće ne formira te krv ne može teći iz desne pretklijetke u klijetku pa u plućnu arteriju. Zbog tog stanja, bebe često razviju i atrijalni septalni defekt ili ventrikularni septalni defekt kako bi se oksigenirana krv pomiješala s deoksigeniranom i na taj način barem djelomično dospjela u mali krvotok.

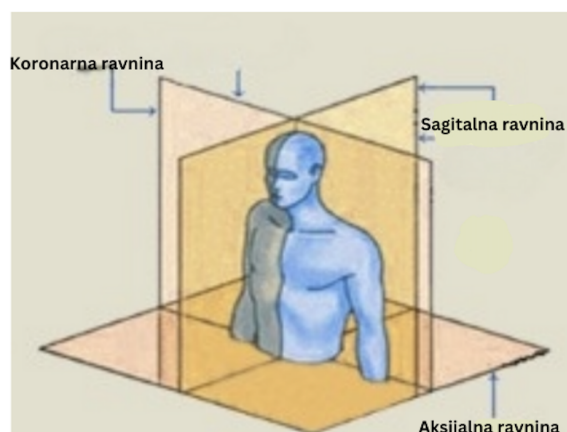
Arterijski trunkus rijetki je srčani defekt kod kojega srce ima jednu glavnu arteriju pa pomiješana oksigenirana i deoksigenirana krv putuje i u pluća i ostatak tijela. Stanje je uobičajeno popraćeno i s ventrikularnim septalnim defektom.

Atrijalni septalni defekt je pojava kada otvor između dvije pretkljetke kroz koji postoji protok krvi. Tijekom trudnoće taj otvor je uobičajen jer omogućava zaobilazak plućnog krvotoka, jer su pluća djeteta tijekom trudnoće nerazvijena. Ako se nakon rođenja prozor ne zatvori, može prouzročiti ozbiljna oštećenja u srcu i plućnoj arteriji. Ipak, značajan dio populacije ima mali preostali otvor koji uobičajeno ne izaziva zdravstvene poteškoće, iako se dovodi vezu s pojavom migrena i moždanog udara. Ta pojava zove se otvoreni foramen ovale. Ventrikularni septalni defekt je stanje u kojem se otvor pojavljuje između kljetki. Ukoliko se otvori na interatrioventrikularnom septumu nalaze i između kljetki i pretkljetki, onda je riječ o atrioventrikularnom septalnom defektu.

4. OBRADA MEDICINSKIH SLIKA

Pomoću medicinskih slika može se kreirati vizualna interpretacija unutrašnjosti tijela sa svrhom dijagnosticiranja i analize medicinskih stanja te planiranja raznih medicinskih zahvata. Najpoznatiji načini dobivanja medicinskih slika su magnetska rezonanca (eng. *magnetic resonance imaging – MRI*), ultrazvuk i CT (eng. *computed tomography*) koji su detaljnije opisani u nastavku.

CT ili računalna tomografija radiološka je metoda pomoću koje se korištenjem ionizirajućeg rendgenskog zračenja dobiva slojeviti prikaz nekog dijela tijela. Princip rada CT-a je temeljen na slabljenju rendgenskih zraka pri prolasku kroz tijelo. Za razliku od klasičnog rendgena koji koristi fiksnu rendgensku cijev, CT koristi rotirajući izvor rendgenskog zračenja oblika prstena dok su rendgenski detektori postavljeni na suprotnom kraju prstena. Slabljenje rendgenskih zraka ovisi o koeficijentu apsorpcije tkiva, odnosno fizikalnom svojstvu tkiva da apsorbira elektromagnetske valove. Tada oslabljeno zračenje pada na detektore koji elektromagnetsko zračenje pretvaraju u električne signale i šalje ih računalu. Nakon svake odrađene rotacije na računalu se tomografskom rekonstrukcijom kreiraju serije dvodimenzionalnih slika (eng. *slices*) iz različitih ravnina, gdje svaka slika obično pokriva područje debljine 1-10mm. Tri su glavne ravnine na kojima se rekonstruira slika: aksijalna, sagitalna i koronarna ravnina. Aksijalna ravnina postavljena je paralelno s tlom i dijeli tijelo na gornji i donji dio. Tako postavljene slike promatraju s vrha ili dna. Sagitalna ravnina okomita je na tlo, a dijeli tijelo na lijevo i desno, a slike se u tom slučaju promatraju sa strane. Koronarna ravnina okomita je na tlo i dijeli tijelo na prednju i stražnju stranu te se promatra sprijeda ili straga [16] (Slika 4.1).



Slika 4.1 Prikaz anatomskih ravnina [17]

4.1. Zapis i obrada medicinskih slika

Medicinska slika prikaz je unutarnje strukture neke od anatomskih regija pomoću polja elemenata koji se nazivaju pikseli. Formati za zapis slika pružaju standardiziran način spremanja informacija o slici u datoteku. Medicinske slike sastoje se od većeg broja slika koje naslagane jedna na drugu mogu predstavljati projekciju trodimenzionalne slike na ekran (projekcija na ravninu), niz slika, gdje svaka slika predstavlja tanki sloj unutar nekog obujma (tomografsko ili višeslojno dvodimenzionalno snimanje) ili više prikaza iste 3D slike u ovisnosti o vremenu u svrhu dobivanja dinamičkog niza 3D slika (4-dimenzionalno snimanje). Format za zapis slika sadrži podatke o tome kako su podaci unutar slike organizirani i kako se pikseli u slici trebaju interpretirati u softveru za pravilno učitavanje i prikaz. Takvi podaci koji sadrže informacije o datoteci nazivaju se meta podaci (podaci o podacima).

Slike su spremljene u računalu kao skup bajtova (eng. *byte*). Svaki bajt sastoji se od 8 bitova. Bit je osnovna jedinica u računarstvu i predstavlja logično stanje s dvije moguće vrijednosti: 0 i 1. Broj bitova s kojima je piksel opisan naziva se dubina piksela ili dubina bita. Ako je slika dimenzija 256 x 256 i ima dubinu od 12 ili 16 bita, znači da je svaki piksel opisan s 2 bajta, te će slika biti veličine 131072 bajtova u oba slučaja. Fotometrijska interpretacija određuje kako će podaci o pikselima biti obrađeni, odnosno je li slika prikazana kao monokromatska ili slika u boji. Ta informacija određena je tzv. kanalima piksela. Monokromatske slike imaju jedan kanal po pikselu gdje je najčešće slika opisana s 8 bitova, pomoću kojih se može dobiti 256 nijansi sive. U tu vrstu obično spadaju CT i magnetska rezonanca. U slučaju PET (eng. *Positron Emission Tomography*) i SPECT (eng. *Single Photon Emission Tomography*) skena slike su prikazane paletom boja, iako je u njihovom slučaju svaka vrijednost mapirana na određenu boju te se boja odnosi jedino na prikaz slike, dok je informacija i dalje opisana jednim kanalom. U tom slučaju radi se o tzv. pseudoboja.

Formati za zapis medicinskih slika mogu se podijeliti u dvije kategorije. Prva kategorija su formati koji služe za standardiziranje dijagnostičkih modaliteta poput CT-a, MRI-a, ultrazvuka itd. Primjer je DICOM (eng. *Digital Imaging and Communications in Medicine*), koji je postao standard za korištenje medicinskih slika. Drugoj kategoriji pripadaju formati čija je svrha potpora za analizu i obradu slike, poput NIFTI, Minc i Analyze formata. Također, medicinske slike mogu se spremati

na dva načina. Prvi je da datoteka sadrži i podatke o slici i meta podatke, gdje su meta podaci uvijek spremljeni u zaglavlju na početku datoteke. Drugi način je da se slika i njeni meta podaci odvoje u dvije datoteke, .hdr i .img, iako je prvi način češći te ga koriste DICOM, NIfTI i Minc.[16]

4.1.1. NIfTI

NIfTI (eng. *The Neuroimaging Informatics Technology Initiative*) je datotečni format kreiran 2000. godine s namjerom stvaranja formata koji će zamijeniti Analyze i ukloniti njegove nedostatke. NIfTI uključuje podršku za unsigned 16-bit tip podatka. Iako podržava dvodatotečnu konfiguraciju s .hdr i .img datotekom, uobičajeno se sprema kao jedinstvena datoteka s ekstenzijom .nii. Iako je u prvom slučaju zaglavlje veličine 348 bajtova, u .nii formatu je veličine 352 bajta kako bi bio višekratnik broja 16, te je moguće koristiti taj prostor za pohranu dodatnih meta podataka. Nova verzija NIfTI standarda NIfTI-2 razvijena je 2011. godine za obradu podataka većih veličina koristeći 64-bitne cijele brojeve u usporedbi sa 16-bitnom starijom verzijom. Novo zaglavlje dolazi sa 544 bajta. NIfTI format podržavaju mnogi vanjski servisi poput 3D Slicera za analizu slike i Nibabel biblioteke za obradu slike u Python programskom jeziku koji su korišteni za potrebe ovog rada.[16]

4.2. Teorijske osnove geometrijskog dubokog učenja

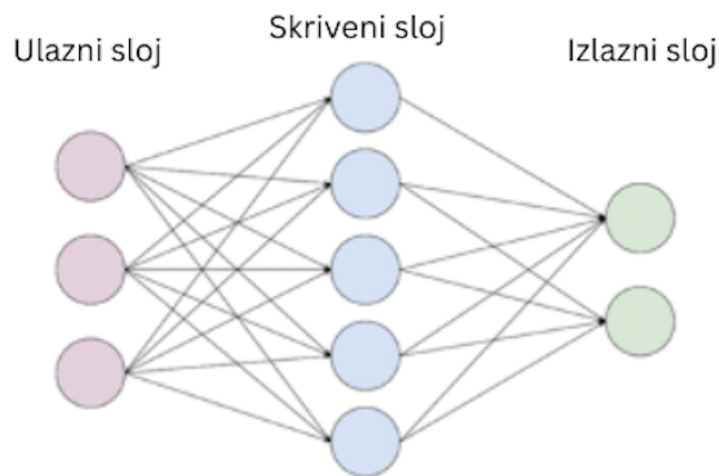
4.2.1. Neuronske mreže

Neuronske mreže, odnosno umjetne neuronske mreže (eng. *artificial neural networks – ANN*) osnova su današnjih algoritama dubokog učenja. Njihov naziv i struktura inspirirani su biološkim neuronskim mrežama u ljudskom mozgu. Kao i biološke, umjetne neuronske mreže uče i ažuriraju znanje, ovisno o okolini kojoj su izloženi te predstavljaju matematički model prirodnog živčanog sustava. Ukratko, to je skup matematičkih algoritama koji, za neki ulazni podatak izvode određene operacije i vraćaju izlaznu vrijednost. Primjena neuronskih mreža vrlo je rasprostranjena u znanosti i uključuje prepoznavanje teksta i govora, algoritme za pretraživanje na webu, algoritme na društvenim mrežama, vremenske prognoze, predviđanje vrijednosti na burzama, segmentacije medicinskih slika i dr..

Neuronske mreže sastavljene su od ulaznog sloja, izlaznog sloja i jednog ili više skrivenih slojeva. Slojevi su predstavljeni čvorovima, a međusobno su spojeni bridovima (Slika 4.2). Ulazni sloj mjesto je gdje vanjska informacija dolazi do neuronske mreže te ju se tamo procesira, analizira i kategorizira, prije nego što ju se proslijedi prema sljedećem sloju. Skriveni slojevi, kojih uobičajeno ima mnogo, transformiraju ulaznu informaciju te rezultat šalju prema izlazu. Svaki čvor u skrivenom sloju sadrži aktivacijsku funkciju koja obrađuje ulaznu informaciju i šalje novu vrijednost prema sljedećem sloju. Bridovi sadrže težine (eng. *weights*) koje označavaju koliko je određena veza između čvorova značajna. Na svako računanje u čvoru dodaje se vrijednost utjecaja (eng. *bias*). Izraz za računanje izlazne vrijednosti je:

$$output = input * weight + bias$$

Model neuronske mreže sposoban je generalizirati problem iz stvarnog svijeta upravo zahvaljujući promjeni vrijednosti težina i biasa sa svakom iteracijom izvršavanja. Taj proces naziva se treniranje [18].



Slika 4.2 Shematski prikaz neuronske mreže

4.2.2. Konvolucijske neuronske mreže

Konvolucijske neuronske mreže (CNN) vrsta su neuronskih mreža često korištena u vizualnom računarstvu, npr. segmentaciji slika ili binarnoj klasifikaciji, a temelje se na konvoluciji. Konvolucija je matematička operacija koja koristi dvije matrice, jednu nad kojom se operacija

izvodi, te drugu poznatu i kao filter ili kernel, koja služi kao multiplikator i prelazi preko prve matrice i obavlja složenu multiplikaciju. Ovim postupkom omogućavaju se mnoge operacije nad slikama poput izoštravanja ili zamućenja slika.

Konvolucijske neuronske mreže uobičajeno su sastavljene od nekoliko bitnijih slojeva:

- Ulazni sloj
- Konvolucijski sloj
- Aktivacijski sloj
- Objedinjujući sloj (eng. *pooling layer*)

Nad ulaznim slojem odvija se proces konvolucije, koristeći kernel ili filter, koji se pomiče preko slike. O tome koliki je pomak između koraka ovisi i veličina izlaza funkcije. Ta vrijednost uobičajeno se naziva iskorak (eng. *stride*). Nad poljima koja se preklapaju između slike i kernela odvija se neka matematička operacija. Poznati primjer kernela je Gaussovo zamućenje, prikazano na slici 4.3. u obliku 3x3 matrice, koji služi za zamućenje slika (Slika 4.4.). Nakon svake operacije konvolucije odvija se aktivacijska funkcija, koja pomaže neuronskoj mreži da nauči nelinearne veze među svojstvima slike, što doprinosi robusnosti neuronske mreže. Najpoznatiji primjeri aktivacijskih funkcija su:

- ReLU

$$f(x) = \max(0, x) \quad (3-1)$$

- Sigmoid

$$\sigma(z) = \frac{1}{1+e^{-z}} \quad (3-2)$$

ReLU aktivacijska funkcija ima i mnoge varijante, od kojih je Leaky ReLU korištena u ovom radu. Opisana je formulom 3-3.

$$f(x) = \max(0.01 * x, x) \quad (3-3)$$

Ono što je specifično za ovu varijantu je da za razliku od klasične ReLU aktivacijske funkcije dozvoljava mali pozitivni gradijent čime se izbjegava problem umirućeg ReLU-a

U objedinjujućem sloju primjenjuje se neki od agregacijskih kernela čime se smanjuju dimenzije slike, odnosno broj parametara u modelu, dok se istovremeno čuvaju bitne značajke slike. Primjeri

agregacijskih funkcija u kernelu su Max funkcija, koja vraća najveći element iz uzorka, Sum, koja zbraja sve vrijednosti u uzorku i Average koja vraća prosječnu vrijednost.

$\frac{1}{16}$	1	2	1
	2	4	2
	1	2	1

Slika 4.3 3x3 matrica Gaussovog zamućenja[19]



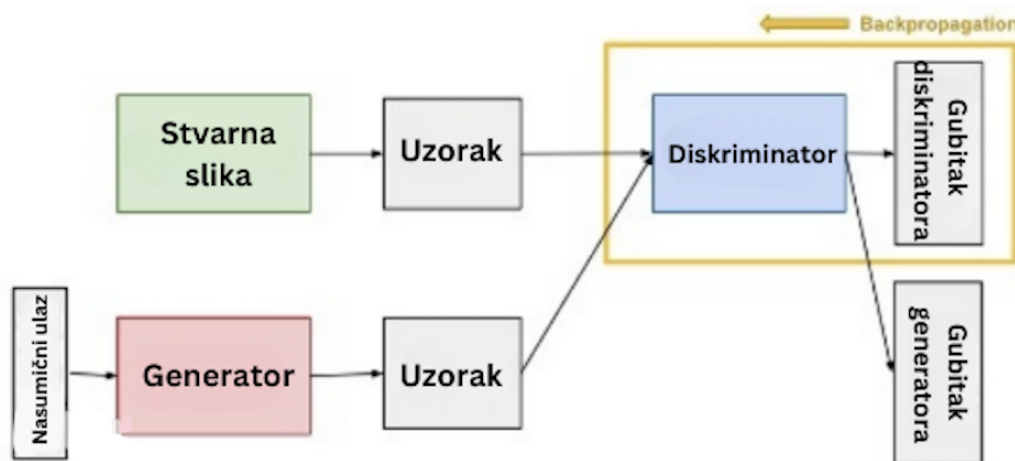
Slika 4.4 Gaussovo zamućenje u ovisnosti o dimenziji kernela[20]

4.2.3. Generative Adversarial Network (GAN)

Generative Adversarial Network - GAN ili generativna kontradiktorna mreža nenadzirani je model strojnog učenja koji je postao bitan čimbenik razvoja segmentacije slika. Model je generativan, tj. kreira vlastite podatke za trening. Temeljen je na teoriji igara, odnosno igri nulte sume u kojoj se međusobno natječu dva modela, generator i diskriminator. Generator je konvolucijska neuronska

mreža čiji je zadatak naučiti kako stvoriti umjetne ulazne podatke koji će zavarati diskriminator. Diskriminator je neuronska mreža koja se trenira da prepozna lažne odnosno stvarne podatke.

Diskriminator se trenira tako da uči prepoznavati lažne podatke koje mu šalje generator, što se u slučaju uspjeha backpropagation algoritmom šalje kao povratna informacija generatoru, koji se u tom slučaju penalizira (Slika 4.5) [21].

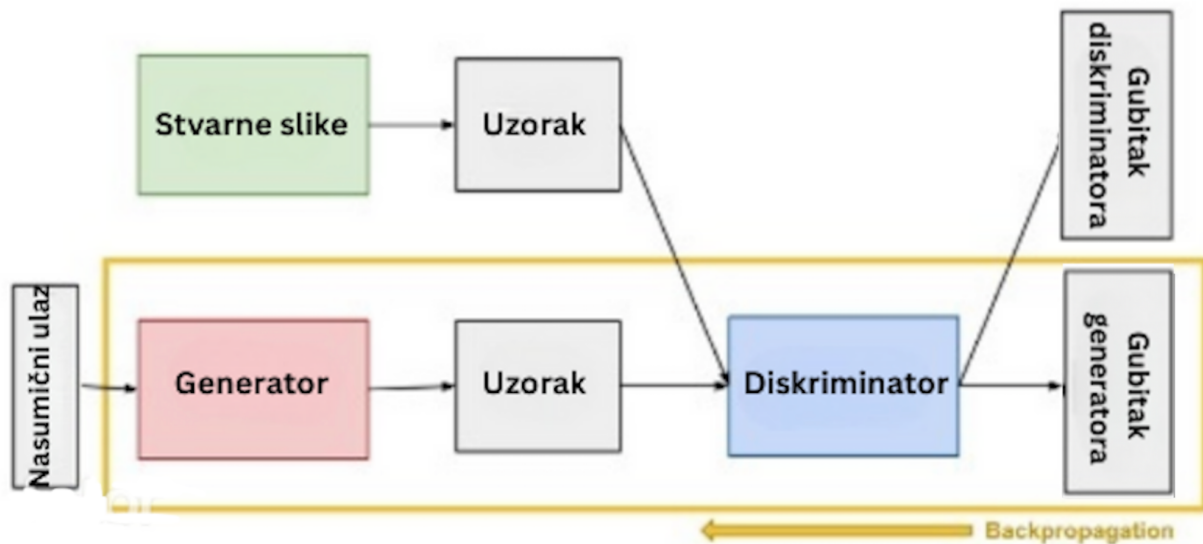


Slika 4.5 Shema treninga diskriminatora [22]

Isto vrijedi i u suprotnom slučaju ako diskriminator ne uspije prepoznati lažne ulazne podatke koje mu šalje generator. Ponavljajući proces u petlji s vremenom će generator stvarati sve uvjerljivije lažne podatke, dok će diskriminator postati bolji u prepoznavanju lažnih podataka.

S druge strane, treniranje generatora zahtijeva veći stupanj integracije između generatora i diskriminatora jer penaliziranje generatora ovisi o evaluaciji diskriminatora kojemu generator šalje ulazne podatke (Slika 4.6). Ulazni podaci za generator su obično nasumična buka. Od nebitne informacije generator sačinjava relevantne trening podatke. U slučaju penalizacije propagira se kroz diskriminator do generatora, ali se gradijent koristi samo za optimiziranje generatora. Razlog za to je što je GAN sačinjen od dvije zasebne neuronske mreže te bi simultano treniranje oba modela potaklo neželjene probleme pri konvergenciji oba modela. Zato se svaki model trenira unakrsno određeni broj epoha, dok je drugi model za to vrijeme konstantan.

Jedan od glavnih problema treniranja GAN modela je prepoznavanje konvergencije. Kada generator napreduje s treningom, diskriminator postaje lošiji u prepoznavanju stvarnih i lažnih podataka. Ako diskriminator ima 50% uspješnost u prepoznavanju, znači da generator radi savršeno. Problem je što diskriminator toga nije svjestan te se generator može početi trenirati nasumičnim podacima iz diskriminatora što bi mu moglo smanjiti kvalitetu. Zbog te promjenjive i nestabilne konvergencije modela, bitno je prepoznati kada je GAN model dobro utreniran.[22]



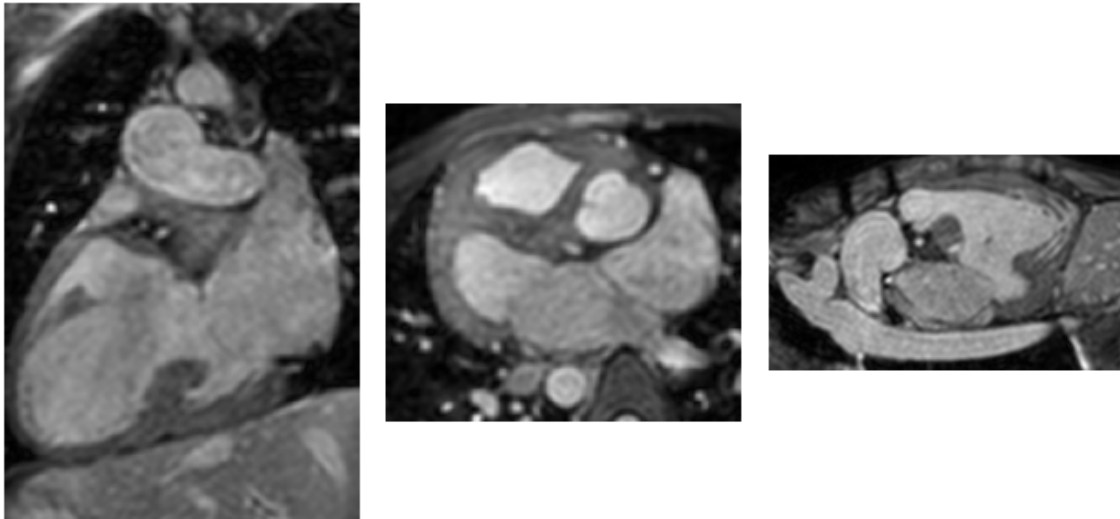
Slika 4.6 Shema treninga generatora [22]

5. SUSTAV ZA SEGMENTACIJU CT SLIKA

Za potrebe segmentacije CT slika korišteno je programsko okruženje Google Colab koje omogućava jednostavno korištenje servisa Jupyter Notebook bez ikakvog postavljanja te pruža računalne resurse uključujući GPU i TPU sredstva. Jupyter Notebook je alat pomoću kojeg se može stvarati dokumente koji sadrže programski kod, jednadžbe, dokumentaciju i vizualne informacije. Korišten je u mnogim područjima, osobito u podatkovnim znanostima, statističkim modelima, vizualizaciji podataka, strojnom učenju itd.

5.1. Razvijeni sustav

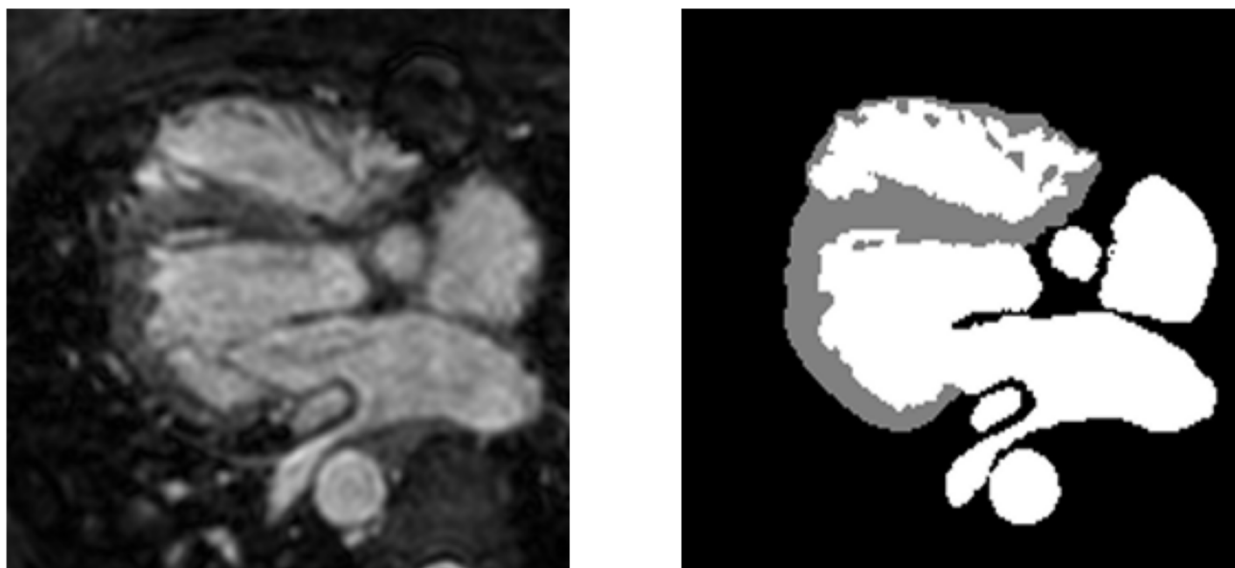
Kao ulazni set podataka korišteno je deset CT slika spremljenih u NIfTI formatu. Sve CT slike imaju i pripadajuće ground truth slike. Ground truth slike stvorili su liječnici i one predstavljaju ispravno segmentirane dijelove srca na temelju kojih se mreža uči (Slika 5.1). Prvi korak koji je potrebno učiniti je pripremiti podatke za obradu, a budući da su takve slike trodimenzionalne potrebno je pretvoriti takvu sliku u set dvodimenzionalnih slika u .png formatu. Implementirana je metoda koja učitava NIfTI datoteke na danoj putanji te s pomoću Nibabel i Numpy biblioteka pretvara dane datoteke u .png slike koje se spremaju u odvojeni direktorij. Važno je pripaziti na tip podataka koji se koristi jer PNG format ne podržava sve formate kao NIfTI. U ovom slučaju korištena je konverzija u uint8 tip podatka. Na slici 3.6 prikazan je presjek jedne od CT slika srca iz koronalne, aksijalne i sagitalne ravnine. U ovom radu koriste se presjeci po aksijalnoj ravnini.



Slika 5.1 Presjek srca iz tri ravnine: koronarne, aksijalne i sagitalne

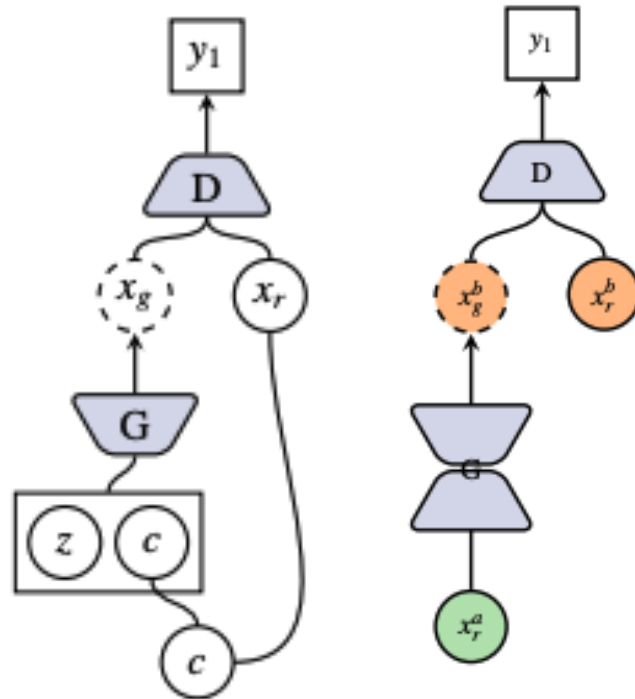
Kao što je vidljivo iz slike 3.6., slike su različitih dimenzija, jer je i sama .nii datoteka sadržavala 3D matricu u nepravilnom obliku. Radi lakše implementacije modela i rada s dostupnim podacima slike treba konvertirati u efikasniji format. Izabran je format 256 x 256 x 3 u kojem slike mogu biti lakše obrađene zbog svojstava broja 256 koji je višekratnik broja 2. Isti proces izveden je i na CT slikama i na njihovoj pripadajućoj ground truth slici (Slika 5.2).

Nakon konverzije podataka, dobivene slike su podijeljene u tri grupe. Prva grupa služi kao skup podataka za trening, gdje slike predstavljaju primjere iz kojih trenirani model uči te prilagođava svoje parametre na temelju performansi. Kako je praćenje performansi ključni aspekt prilikom treniranja GAN modela, testni podaci se koriste za evaluaciju modela nakon svake epohe treninga. Testiranje na tim podacima pruža uvid u trenutno stanje modela, ali ne utječe na njegove parametre jer se ti podaci ne koriste za daljnje učenje. Treća grupa je validacijska, koja se primjenjuje nakon završetka treninga kako bi se provjerila ukupna funkcionalnost modela. Podaci su podijeljeni u omjeru 70:20:10, pri čemu je 70% slika korišteno za trening, 20% za testiranje tijekom procesa treninga, a 10% za validaciju modela nakon što je proces treniranja završen.



Slika 5.2 Primjer CT i GT slike

U radu je korišten model Pix2Pix [23] [24]. Pix2Pix arhitektura definirana je U-Net generatorom i konvolucijskim PatchGAN diskriminatorom koji služi za klasifikaciju slike. Model dubokog učenja temelji se na uvjetnim generativnim kontradiktornim mrežama (eng. *Conditional Generative Adversarial Network – CGAN*). Iako se uobičajeno GAN koristi za generiranje slika kao i za mapiranje slike na sliku, GAN se primjenjuje bezuvjetno oslanjajući se na druge faktore poput L2 regresije kako bi prisilili provjeru izlaznih podataka na ulazu. U slučaju bezuvjetnog GAN-a, generator pokušava prevariti diskriminator, dok diskriminator pokušava shvatiti je li ponuđena slika prava ili je generirana. Kod uvjetnog GAN-a diskriminator ima zadatak otkriti koja je slika lažna između originalne slike i one koju je kreirao generator (Slika 5.3) [25].



Slika 5.3 Shema CGAN (lijevo) i njene Pix2Pix varijacije (desno) [26]

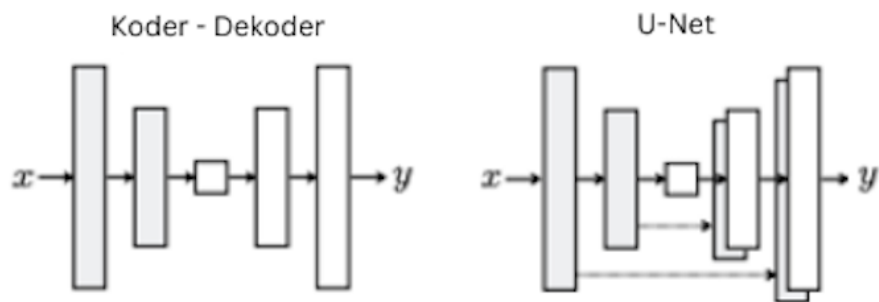
Cilj uvjetnog GAN-a može se opisati formulom 4-1:

$$L_{cGAN}(G, D) = E_{x,y}[\log D(x, y)] + E_{x,z}[\log(1 - D(x, G(x, z)))] \quad (4-1)$$

gdje G pokušava minimizirati vrijednost nasuprot D kojem je cilj maksimizirati ju. Za usporedbu, formula u kojoj se koristi bezuvjetni GAN nalik je jednadžbi 4-2.

$$L_{GAN}(G, D) = E_y[\log D(y)] + E_{x,z}[\log(1 - D(G(x, z)))] \quad (4-2)$$

Generator se u ovom slučaju temelji na U-Net arhitekturi. U-Net je posebna vrsta konvolucijskih neuronskih mreža koja se sastoji od dva glavna dijela: kontrakcijskog i ekspanzivnog. Iako je slična koder-dekoder arhitekturi, razlikuje se po vezama koje postoje između zrcalnih slojeva (Slika 5.4).



Slika 5.4 Koder-dekoder i U-Net [7]

U takvom modelu ulazna slika prolazi kroz niz slojeva kroz koji se progresivno poduzorkuje, sve do bottleneck sloja. Za mnoge probleme transliranja slika postoji velika količina informacija niske razine koje povezuju ulaznu i izlaznu sliku te je poželjno propagirati te informacije kroz neuronsku mrežu. Primjer kod segmentacije je lokacija izloženih rubova koja je jednaka za ulaz i izlaz. Kako bi se informacija mogla proslijediti kroz mrežu dodaju se preskokne veze, kao što je prikazano i na slici 5.4. Veze se dodaju između svakog i -tog i $(n-i)$ -tog sloja gdje je n ukupni broj slojeva [7].

Iz koda su vidljivi slojevi poduzorkovanja (Slika 5.5) gdje se smanjuju prostorne dimenzije duljine i širine slike dok im se povećava dubina, odnosno broj kanala. Leaky ReLU aktivacijska funkcija omogućava malim negativnim vrijednostima prolazak čime se rješava problem umirućeg ReLU-a. Kod naduzorkovanja (Slika 5.6) povećavaju se dimenzije slike, i primjenjuje se dropout u početnim fazama kako bi se spriječio overfitting ulazne i izlazne slike, odnosno pojava kada je izlazni podatak jako sličan ili identičan trening podacima što izrazito negativno utječe na testiranje modela. Vjerojatnost za dropout je u tim slučajevima 0.5. Tijekom naduzorkovanja kao parametar se prosljeđuju slojevi poduzorkovanja čime se postiže simetrična veza između kontrakcijskog i ekspanzivnog dijela. U konačnici se rekonstruira drugačija slika koristeći svojstva ulazne slike.

```

class Downsample(tf.keras.Model):
    def __init__(self, filters, size, apply_batchnorm=True):
        super(Downsample, self).__init__()
        self.apply_batchnorm = apply_batchnorm
        initializer = tf.random_normal_initializer(0., 0.02)

        self.conv1 = tf.keras.layers.Conv2D(filters,
                                             (size, size),
                                             strides=2,
                                             padding='same',
                                             kernel_initializer=initializer,
                                             use_bias=False)

        if self.apply_batchnorm:
            self.batchnorm = tf.keras.layers.BatchNormalization()

    def call(self, x, training):
        x = self.conv1(x)
        if self.apply_batchnorm:
            x = self.batchnorm(x, training=training)
        x = tf.nn.leaky_relu(x)
        return x

```

Slika 5.5 Klasa Downsample

```

class Upsample(tf.keras.Model):
    def __init__(self, filters, size, apply_dropout=False):
        super(Upsample, self).__init__()
        self.apply_dropout = apply_dropout
        initializer = tf.random_normal_initializer(0., 0.02)

        self.up_conv = tf.keras.layers.Conv2DTranspose(filters,
                                                       (size, size),
                                                       strides=2,
                                                       padding='same',
                                                       kernel_initializer=initializer,
                                                       use_bias=False)

        self.batchnorm = tf.keras.layers.BatchNormalization()
        if self.apply_dropout:
            self.dropout = tf.keras.layers.Dropout(0.5)

    def call(self, x1, x2, training):
        x = self.up_conv(x1)
        x = self.batchnorm(x, training=training)
        if self.apply_dropout:
            x = self.dropout(x, training=training)
        x = tf.nn.relu(x)
        x = tf.concat([x, x2], axis=-1)
        return x

```

Slika 5.6 Klasa Upsample

```

self.down1 = Downsample(64, 4, apply_batchnorm=False)
self.down2 = Downsample(128, 4)
self.down3 = Downsample(256, 4)
self.down4 = Downsample(512, 4)
self.down5 = Downsample(512, 4)
self.down6 = Downsample(512, 4)
self.down7 = Downsample(512, 4)
self.down8 = Downsample(512, 4)

self.up1 = Upsample(512, 4, apply_dropout=True) # to prevent overfitting
self.up2 = Upsample(512, 4, apply_dropout=True)
self.up3 = Upsample(512, 4, apply_dropout=True)
self.up4 = Upsample(512, 4)
self.up5 = Upsample(256, 4)
self.up6 = Upsample(128, 4)
self.up7 = Upsample(64, 4)

self.last = tf.keras.layers.Conv2DTranspose(OUTPUT_CHANNELS,
                                             (4, 4),
                                             strides=2,
                                             padding='same',
                                             kernel_initializer=initializer)

```

Slika 5.7 Poziv generatora

```

def call(self, x, training):
    # x shape == (batch_size, 256, 256, 3)
    x1 = self.down1(x, training=training) # (batch_size, 128, 128, 64)
    x2 = self.down2(x1, training=training) # (batch_size, 64, 64, 128)
    x3 = self.down3(x2, training=training) # (batch_size, 32, 32, 256)
    x4 = self.down4(x3, training=training) # (batch_size, 16, 16, 512)
    x5 = self.down5(x4, training=training) # (batch_size, 8, 8, 512)
    x6 = self.down6(x5, training=training) # (batch_size, 4, 4, 512)
    x7 = self.down7(x6, training=training) # (batch_size, 2, 2, 512)
    x8 = self.down8(x7, training=training) # (batch_size, 1, 1, 512)

    x9 = self.up1(x8, x7, training=training) # (batch_size, 2, 2, 1024)
    x10 = self.up2(x9, x6, training=training) # (batch_size, 4, 4, 1024)
    x11 = self.up3(x10, x5, training=training) # (batch_size, 8, 8, 1024)
    x12 = self.up4(x11, x4, training=training) # (batch_size, 16, 16, 1024)
    x13 = self.up5(x12, x3, training=training) # (batch_size, 32, 32, 512)
    x14 = self.up6(x13, x2, training=training) # (batch_size, 64, 64, 256)
    x15 = self.up7(x14, x1, training=training) # (batch_size, 128, 128, 128)

    x16 = self.last(x15) # (batch_size, 256, 256, 3)
    x16 = tf.nn.tanh(x16)

    return x16

```

Slika 5.8 Generator s poduzorkovanjem i naduzorkovanjem

Kao diskriminator (Slika 5.9) koristi se PatchGAN arhitektura. PatchGAN diskriminator pokušava klasificirati je li određeni dio slike (eng. *patch*) stvaran ili ne. U diskriminatoru dijelovi slike se

poduzorkuju, čime se dimenzije slike smanjuju sa 256 x 256 na 32 x 32. Nakon toga dodaje se padding s nulama kako bi se održale dimenzije slike nakon konvolucijskog sloja. Provodi se batch normalizacija, proces koji je standardan u ovakvim modelima i koji normalizira podatke što povećava njihovu korisnost i ubrzava efikasnost gradijentnog spusta. Gradijentni spust (eng. *gradient descent*) ključna je funkcija u modeliranju neuronskih mreža kojoj je cilj minimizacija funkcije gubitka, što se postiže postupnom optimizacijom parametara modela. Kao aktivacijska funkcija korištena je Leaky ReLU funkcija zbog svojih prednosti kod smanjenja umirućeg ReLU-a omogućavanjem blagog gradijentnog uspona. S pozivom last metode diskriminator donosi odluku o validnosti ulazne slike.

```
class Discriminator(tf.keras.Model):
    def __init__(self):
        super(Discriminator, self).__init__()
        initializer = tf.random_normal_initializer(0., 0.02)

        self.down1 = DiscDownsample(64, 4, False)
        self.down2 = DiscDownsample(128, 4)
        self.down3 = DiscDownsample(256, 4)

        # (bs, 32, 32, 256) -> (bs, 31, 31, 512)
        self.zero_pad1 = tf.keras.layers.ZeroPadding2D()
        self.conv = tf.keras.layers.Conv2D(512,
                                           (4, 4),
                                           strides=1,
                                           kernel_initializer=initializer,
                                           use_bias=False)
        self.batchnorm1 = tf.keras.layers.BatchNormalization()
        # (bs, 31, 31, 512) -> (bs, 30, 30, 1)
        self.zero_pad2 = tf.keras.layers.ZeroPadding2D()
        self.last = tf.keras.layers.Conv2D(1,
                                           (4, 4),
                                           strides=1,
                                           kernel_initializer=initializer)

    def call(self, inp, tar, training):
        x = tf.concat([inp, tar], axis=-1)
        x = self.down1(x, training=training) # (bs, 128, 128, 64)
        x = self.down2(x, training=training) # (bs, 64, 64, 128)
        x = self.down3(x, training=training) # (bs, 32, 32, 256)

        x = self.zero_pad1(x) # (bs, 34, 34, 256)
        x = self.conv(x) # (bs, 31, 31, 512)
        x = self.batchnorm1(x, training=training)
        x = tf.nn.leaky_relu(x)

        x = self.zero_pad2(x) # (bs, 33, 33, 512)
        x = self.last(x) # (bs, cen, 30, 1 )

        return x
```

Slika 5.9 Diskriminator

Funkcija gubitka matematička je funkcija koja mjeri koliko dobro model strojnog učenja funkcionira na danom skupu podataka. Funkcija kvantificira razliku između predviđenog izlaza i stvarnog izlaza u obliku numeričke vrijednosti koja predstavlja grešku. Budući da generator i diskriminator imaju potpuno drugačiju svrhu, gdje generator generira nove slike, dok diskriminator služi isključivo klasifikaciji slika, njihove funkcije gubitka drugačijeg su karaktera i očekivano je da imaju drugačiji raspon vrijednosti. Funkcija gubitka za generator (Slika 5.10) koristi sigmoid cross entropy funkciju za izračunavanje gubitka te srednju apsolutnu pogrešku (L1 gubitak) između generiranog izlaza i željenog izlaza s vrijednosti $\lambda = 100$ [7].

```
def generator_loss(generated_discriminator_output, generator_output, target):
    gan_loss = tf.compat.v1.losses.sigmoid_cross_entropy(multi_class_labels = tf.ones_like(generated_discriminator_output),
                                                         logits = generated_discriminator_output)

    l1_loss = tf.reduce_mean(tf.abs(target - generator_output))

    total_gen_loss = gan_loss + (LAMBDA * l1_loss)

    return total_gen_loss
```

Slika 5.10 Funkcija gubitka generatora

Funkcija gubitka diskriminatora (Slika 5.11) prima dva ulaza, stvarnu i generiranu sliku i vraća zbroj gubitaka za stvarni i generirani ulaz, gdje se za stvarni ulaz koristi matrica jedinica, a za generirani ulaz nul-matrica.

```
def discriminator_loss(real_discriminator_output, generated_discriminator_output):
    real_loss = tf.compat.v1.losses.sigmoid_cross_entropy(multi_class_labels = tf.ones_like(real_discriminator_output),
                                                         logits = real_discriminator_output)

    generated_loss = tf.compat.v1.losses.sigmoid_cross_entropy(multi_class_labels = tf.zeros_like(generated_discriminator_output),
                                                             logits = generated_discriminator_output)

    total_loss = real_loss + generated_loss

    return total_loss
```

Slika 5.11 Funkcija gubitka diskriminatora

Uz funkciju gubitka, još jedna bitna komponenta modela dubokog učenja je i optimizacijska funkcija, čiji je cilj modificirati parametre modela kako bi minimizirao funkciju gubitka i time postigao najveću šansu za generiranje preciznih predviđanja. U radu je korišten Adam (eng.

Adaptive Moment Estimation) optimizator koji je postao jedan od najkorištenijih optimizacijskih funkcija u dubokom učenju. Korišteni su parametri $\beta_1 = 0.5$, $\varepsilon = 0.0002$, $\beta_2 = 0.999$.

5.2.Rezultati

Skup podataka podijeljen je na skup za treniranje, skup za testiranje te za validaciju. Provedeno je treniranje modela na skupu trening podataka od 1203 slike dobivene iz CT snimki srca. Trening je proveden na maksimalno 70 epoha, zbog ograničenosti resursa i vremena. Tijekom treninga, između svake epohe model je testiran na skupu za testiranje, kako bi se pratio napredak treninga. Nekoliko primjera prikazano je na sljedećim slikama (Slike 5.12 – 5.14). S lijeve strane prikazane su CT slike srca iz sagitalne ravnine, u sredini su GT slike, odnosno segmentacije srca, dok su s desne strane prikazane predikcije treniranog modela.



Slika 5.12 CT slika (lijevo), GT slika (desno) i predikcija modela (desno)



Slika 5.13 CT slika (lijevo), GT slika (desno) i predikcija modela (desno)



Slika 5.14 CT slika (lijevo), GT slika (desno) i predikcija modela (desno)

Iz prikazanih primjera vidljivo je da, iako je model uspio do jedne mjere naučiti segmentirati CT slike, prisutno je zamućenje slike, kao i nedovoljna točnost. Razlozi za loše performanse modela uključuju dizajn modela, vrijeme treniranja i količinu podataka za trening i kvalitetu podataka. Zbog same složenosti GAN modela, izvođenje treninga izuzetno je zahtjevno i vremenski i fizički. Bolji rezultati mogli bi se postići trenirajući model s nekom od jačih grafičkih kartica s izrazito visokim performansama, budući da je optimalan broj epoha za trening modela veći od 70, kao npr. 200, vrijednost koja je određena u [7]. Zamućenje prisutno kod slika može se preispitati korištenjem drugačijeg kernela pri konvoluciji, ili promjenom drugih parametara modela. S druge strane, u sljedećim slikama (Slike 5.15, 5.16) prikazani su rezultati modela nakon treniranja na CT slikama iz aksijalne ravnine. Za razliku od prošlog slučaja, ovdje je konverzija podataka iz 3D u 2D slike pomoću vanjskog alata za konverziju čiji izvorni kod nije javno dostupan. Vidljivo je da su rezultati bolji u odnosu na prethodne što implicira da je u prethodnom slučaju problem i kod kvalitete slike, koja je izgubila kvalitetu pri ručnoj konverziji 3D slike.



Slika 5.15 Sagitalna ravnina, CT slika (lijevo), GT slika (desno) i predikcija modela (desno)



Slika 5.16 Sagitalna ravnina, CT slika (lijevo), GT slika (desno) i predikcija modela (desno)

6. ZAKLJUČAK

U radu je opisana funkcija srca u ljudskom tijelu te su opisane najpoznatije urođene srčane mane i njihova podjela. Predstavljen je način obrade medicinskih CT slika i opisane su teorijske osnove neuronskih mreža, potrebnih za implementaciju programskog rješenja. Implementirana je metoda za segmentaciju srca iz 3-dimenzionalnih CT slika u NIfTI formatu. 3D slike konvertirane su u skupove 2D slika u aksijalnoj ravnini za obradu. Segmentacija srca napravljena je korištenjem modela dubokog učenja koristeći GAN arhitekturu. Generator je implementiran koristeći U-Net, a diskriminator koristeći PatchGAN arhitekturu. Rezultati segmentacije pokazuju na veliku mogućnost poboljšanja, prije svega optimizacijom pri konverziji podataka kako bi se smanjili gubici ključnih informacija, te korištenjem više računalnih resursa kako bi bilo moguće trenirati model na podacima veće kvalitete kroz znatno veći broj iteracija čime bi se postigle bolje performanse.

LITERATURA

- [1] Rim B, Lee S, Lee A, Gil HW, Hong M. Semantic Cardiac Segmentation in Chest CT Images Using K-Means Clustering and the Mathematical Morphology Method. *Sensors (Basel)*. 2021 Apr 10;21(8):2675. doi: 10.3390/s21082675. PMID: 33920219; PMCID: PMC8070040.
- [2] B.E.D. Helmy, *Image Processing: Graph-based Segmentation*, Baeldung, 2023. dostupno na: <https://www.baeldung.com/cs/graph-based-segmentation> [1.9.2024.]
- [3] P. Roy, S. Dutta, N. Dey, G. Dey, S. Chakraborty and R. Ray, "Adaptive thresholding: A comparative study," 2014 International Conference on Control, Instrumentation, Communication and Computational Technologies (ICCICCT), Kanyakumari, India, 2014, pp. 1182-1186, doi: 10.1109/ICCICCT.2014.6993140.
- [4] B.E.D. Helmy, *Understanding Otsu's Method for Image Segmentation*, Baeldung, 2023, dostupno na: <https://www.baeldung.com/cs/otsu-segmentation> [1.9.2024.]
- [5] Wenqi Ji, Xiaoguang He. Kapur's entropy for multilevel thresholding image segmentation based on moth-flame optimization[J]. *Mathematical Biosciences and Engineering*, 2021, 18(6): 7110-7142.
- [6] Image Tresholding, OpenCV Documentation, dostupno na: https://docs.opencv.org/4.x/d7/d4d/tutorial_py_thresholding.html [1.9.2024.]
- [7] P. Isola, J.Y. Zhu, T. Zhou, A.A. Efros, Image-to-image translation with conditional adversarial networks. 2017, In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1125-1134).
- [8] S. Bruns, J.M. Wolterink, T.P.W. van den Boogert, J.H. Runge, B.J. Bouma, J.P. Henriques, J.Baan, M.A. Viergever, R. Nils Planken, I. Išgum, Deep learning-based whole-heart segmentation in 4D contrast-enhanced cardiac CT, *Computers in Biology and Medicine*, Volume 142, 2022, 105191, ISSN 0010-4825
- [9] A.J. Weinhaus, K.P. Roberts, *Anatomy of the Human Heart*, 2005, In: Iaizzo, P.A. (eds) *Handbook of Cardiac Anatomy, Physiology, and Devices*. Humana Press. https://doi.org/10.1007/978-1-59259-835-9_4
- [10] Srce, Wikipedija, dostupno na: <https://hr.wikipedia.org/wiki/Srce> [17.9.2024.]
- [11] How does the blood circulatory system work?, Institute for Quality and Efficiency in Health Care, Koln, Njemačka, 2023, dostupno na: <https://www.ncbi.nlm.nih.gov/books/NBK279250/> [1.9.2024.]
- [12] About Congenital Heart Defects (CHDs), U.S. Centers for Disease Control and Prevention, 2024. dostupno na: <https://www.cdc.gov/ncbddd/heartdefects/facts.html> [1.9.2024.]

- [13] Common Types of Heart Defects, American Heart Association, dostupno na: <https://www.heart.org/en/health-topics/congenital-heart-defects/about-congenital-heart-defects/common-types-of-heart-defects> [1.9.2024.]
- [14] A. Gregov, "Analiza deformacije miokarda magnetskom rezonancijom u bolesnika sa sistemskom desnom klijetkom", Diplomski rad, Sveučilište u Zagrebu, Medicinski fakultet, Zagreb, 2022. Dostupno na: <https://urn.nsk.hr/urn:nbn:hr:105:742567>
- [15] J.A. Dearani, B.N. Mora, T.J. Nelson, D.T. Haile & P.W. O'Leary, Ebstein anomaly review: what's now, what's next?, Expert Review of Cardiovascular Therapy, vol. 13,10 (2015): 1101-9.
- [16] M.Larobina, L.Murino, Medical Image File Formats, 2013, Journal of digital imaging. 10.1007/s10278-013-9657-9.
- [17] Image Reconstruction Planes, Boehringer Ingelheim, 2023, dostupno na: <https://www.ipfradiologyrounds.com/hrct-primer/image-reconstruction/> [1.9.2024.]
- [18] A. D'Agostino, Introduction to neural networks — weights, biases and activation, Medium, 2021, dostupno na: <https://medium.com/@theDrewDag/introduction-to-neural-networks-weights-biases-and-activation-270ebf2545aa> [1.9.2024.]
- [19] R.Lahiri, Gaussian Blurring and it's importance in Image Processing, Medium, 2020., dostupno na: <https://medium.com/@rlahiri/gaussian-blurring-and-its-importance-in-image-processing-4be8915b85ec> [1.9.2024.]
- [20] P. Antoniadis, How Do Blurs in Images Work?, Baeldung, 2024. dostupno na: <https://www.baeldung.com/cs/cv-blur-images> [1.9.2024.]
- [21] K.Yasar, Generative Adversarial Network (GAN), TechTarget, 2023. dostupno na: <https://www.techtarget.com/searchenterpriseai/definition/generative-adversarial-network-GAN> [1.9.2024.]
- [22] GAN Introduction, Google for Developers, 2022, dostupno na: <https://developers.google.com/machine-learning/gan> [1.9.2024.]
- [23] pix2pix: Image-to-image translation with a conditional GAN, Tensorflow, 2024. dostupno na: <https://www.tensorflow.org/tutorials/generative/pix2pix> [1.9.2024.]
- [24] Ground Truth Images Segmentation with GAN-based Model, Mejdoul Z., GitHub, dostupno na: <https://github.com/zakariamejdoul/ground-truth-gan-segmentation> [1.9.2024.]

[25] W.Li, Y.Li, W.Qin, X.Liang, J.Xu, J.Xiong, Y.Xie, Magnetic resonance image (MRI) synthesis from brain computed tomography (CT) images based on deep learning methods for magnetic resonance (MR)-guided radiotherapy. *Quantitative Imaging in Medicine and Surgery*. 2020, 10. 1223-1236. 10.21037/qims-19-885.

[26] X.Yi, E.Walia, P.Babyn, Generative adversarial network in medical imaging: A review, *Medical Image Analysis*, Volume 58, 2019, 101552, ISSN 1361-8415

SAŽETAK

Kardiovaskularne bolesti vodeći su uzrok smrti danas u svijetu. Urođene srčane mane jedan su od faktora koji doprinosi toj statistici. Razvojem znanosti došlo je do napretka metoda obrade digitalnih slika. Razvijene su metode segmentacije dijelova srca temeljene na dubokom učenju. U ovom je radu predstavljena metoda segmentacije srca iz 3D CT slika koristeći GAN metodu dubokog učenja. Za potrebe treninga GAN modela korišteno je 7 CT snimki, konvertiranih u 1203 dvodimenzionalne slike. Prikazani su rezultati treninga modela i usporedba treninga slika iz različitih ravnina, dobivenih različitim načinima konverzije te su dani prijedlozi za poboljšanje performansi modela. U radu je prikazano da se korištenjem GAN arhitekture mogu postići uspješni rezultati u segmentaciji urođenih srčanih mana korištenjem relativno malog skupa podataka za trening.

Ključne riječi

GAN, konvolucijske neuronske mreže, računalna tomografija, segmentacija slike, urođene srčane mane

ABSTRACT

Cardiovascular diseases are the leading cause of death in the world today. Congenital heart defects are one of the factors that contributes to that statistic. The development of science has led to the advancement of digital image processing methods. Heart segmentation methods based on deep learning have been developed. This paper presents a heart segmentation method from 3D CT images using GAN architecture. For the purposes of training the GAN model, 7 CT scans were converted into 1203 two-dimensional images. The results of model training and a comparison of training images from different planes, obtained by different conversion methods, are presented, and suggestions for improving the model's performance are given. The paper shows that by using the GAN architecture, successful results can be achieved in the segmentation of congenital heart defects using a relatively small set of training data.

Keywords

computer tomography, congenital heart defects, convolutional neural networks, GAN, image segmentation