

# Generativne suparničke mreže za umjetno generiranje medicinskih slika

---

Gavran, Iva

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:200:199851>

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

*Download date / Datum preuzimanja:* **2024-12-28**

*Repository / Repozitorij:*

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



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU**

**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I  
INFORMACIJSKIH TEHNOLOGIJA**

**Sveučilišni studij**

**GENERATIVNE SUPARNIČKE MREŽE ZA UMJETNO  
GENERIRANJE MEDICINSKIH SLIKA**

**Diplomski rad**

**Iva Gavran**

**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**

<b>Ime i prezime pristupnika:</b>	Iva Gavran
<b>Studij, smjer:</b>	Sveučilišni diplomski studij Računarstvo
<b>Mat. br. pristupnika, god.</b>	D-1201R, 07.10.2021.
<b>JMBAG:</b>	0130311410
<b>Mentor:</b>	prof. dr. sc. Irena Galić
<b>Sumentor:</b>	dr. sc. Marija Habijan
<b>Sumentor iz tvrtke:</b>	
<b>Predsjednik Povjerenstva:</b>	prof. dr. sc. Krešimir Nenadić
<b>Član Povjerenstva 1:</b>	dr. sc. Marija Habijan
<b>Član Povjerenstva 2:</b>	doc. dr. sc. Krešimir Romić
<b>Naslov diplomskog rada:</b>	Generativne suparničke mreže za umjetno generiranje medicinskih slika
<b>Znanstvena grana diplomskog rada:</b>	<b>Umjetna inteligencija (zn. polje računarstvo)</b>
<b>Zadatak diplomskog rada:</b>	Istražiti i opisati način dobivanja medicinskih CT i MRI slika i njihove najvažnije karakteristike. Objasniti teorijske osnove obrade medicinskih slika neuronskim mrežama, tipove i najčešće korištene modele. Dati kratak pregled područja prethodnih istraživanja. Istražiti i opisati problematiku nedostatka medicinskih podataka za učenje neuronskih mreža. Istražiti i opisati generativne suparničke mreže. Razviti sustav za sintetičko generiranje originalnih i pripadnih anotiranih CT i MRI slika. Prikazati i objasniti rezultate te odrediti preciznost izvođenja
<b>Datum ocjene pismenog dijela diplomskog rada od strane mentora:</b>	23.09.2024.
<b>Ocjena pismenog dijela diplomskog rada od strane mentora:</b>	Izvrstan (5)
<b>Datum obrane diplomskog rada:</b>	1. 10. 2024
<b>Ocjena usmenog dijela diplomskog rada (obrane):</b>	Izvrstan (5)
<b>Ukupna ocjena diplomskog rada:</b>	Izvrstan (5)
<b>Datum potvrde mentora o predaji konačne verzije diplomskog rada čime je pristupnik završio sveučilišni diplomski studij:</b>	07.10.2024.



**FERIT**

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

## IZJAVA O IZVORNOSTI RADA

Osijek, 07.10.2024.

**Ime i prezime Pristupnika:**

Iva Gavran

**Studij:**

Sveučilišni diplomski studij Računarstvo

**Mat. br. Pristupnika, godina upisa:**

D-1201R, 07.10.2021.

**Turnitin podudaranje [%]:**

15

Ovom izjavom izjavljujem da je rad pod nazivom: **Generativne suparničke mreže za umjetno generiranje 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

<b>1. UVOD</b> .....	1
<b>1.1. Zadatak diplomskog rada</b> .....	2
<b>2. PREGLED PODRUČJA I PROBLEMATIKE</b> .....	3
<b>3. MEDICINSKE CT I MRI SNIMKE</b> .....	8
<b>3.1. Računalna tomografija</b> .....	8
<b>3.2. Magnetska rezonancija</b> .....	10
<b>4. OBRADA MEDICINSKIH SLIKA NEURONSKIM MREŽAMA</b> .....	11
<b>4.1. Neuronska mreža</b> .....	11
<b>4.1.1. Duboka neuronska mreža</b> .....	12
<b>4.2. Primjena neuronskih mreža u obradi medicinskih slika</b> .....	13
<b>4.2.1. Klasifikacija, segmentacija i detekcija objekta</b> .....	13
<b>4.2.2. Rekreiranje, registracija te generiranje slika</b> .....	14
<b>4.3. Generativne suparničke mreže</b> .....	15
<b>4.3.1. Nastanak i razvoj generativnih suparničkih mreža</b> .....	16
<b>4.3.2. Izazovi i poteškoće prilikom rada s GAN-ovima</b> .....	16
<b>4.3.3. Treniranje generativnih suparničkih mreža</b> .....	17
<b>5. MODEL GAN-A ZA GENERIRANJE MEDICINSKIH SNIMAKA I PRIPADNIH ANOTIRANIH MASKI</b> .....	20
<b>5.1. Podatkovni skup</b> .....	20
<b>5.2. Razvojno okruženje i korištene biblioteke</b> .....	23
<b>5.2.1. Razvojno okruženje</b> .....	23
<b>5.2.2. Ključne biblioteke</b> .....	23
<b>5.3. Pretvaranje snimaka i maski u NumPy nizove i normalizacija maski</b> .....	24

<b>5.4. Predprocesiranje podataka .....</b>	<b>25</b>
<b>5.5. Generator .....</b>	<b>26</b>
<b>5.6. Diskriminator .....</b>	<b>28</b>
<b>5.7. Funkcija gubitka i predtreniranje diskriminatora .....</b>	<b>30</b>
<b>5.8. Treniranje modela i krajnji rezultat .....</b>	<b>31</b>
<b>6. ZAKLJUČAK.....</b>	<b>35</b>
<b>LITERATURA .....</b>	<b>36</b>

# 1. UVOD

Napredak u umjetnoj inteligenciji ne bi bio moguć bez pristupa kvalitetnim skupovima podataka. Podatci čine osnovu dubokog učenja, a kvaliteta i količina tih podataka ključni su za postizanje uspješnih rezultata. U specifičnim situacijama, poput medicine, izrazito je važno posjedovati dovoljno kvalitetnih snimaka jer u suprotnom neće biti moguće razviti učinkovite modele dubokog učenja koji primjerice detektiraju ili prognoziraju bolesti. Međutim, prikupiti dovoljno velik skup medicinskih snimaka popriličan je izazov, prvenstveno zbog očuvanja privatnosti pacijenata, ali i zato što su snimke određenih bolesti izrazito rijetke.

Jedno od mogućih rješenja ovog problema je generiranje umjetnog sadržaja koji je sličan sadržaju podatkovnog skupa. Ovaj pristup moguće je ostvariti korištenjem generativne suparničke mreže (GAN). Posljednjih godina, napredak u području umjetne inteligencije doživio je značajan razvoj zahvaljujući upravo ovom modelu dubokog učenja. Model generativnih suparničkih mreža radi na principu konstantnog sukobljavanja dviju mreža, generatora i diskriminatora. Kroz ovaj iterativni proces, mreže se međusobno poboljšavaju, što dovodi do generiranja novog, realističnog sadržaja.

Zanimljivo je primijetiti da se taj proces može usporediti i s načinom na koji ljudi uče. Primjerice, novorođenče prvotno izgovara glasove nalik na riječi, a roditelji ga ispravljaju sve dok ne nauči izgovoriti nešto razumljivo. S vremenom dijete, osim što nauči izgovarati pojedinačne riječi, počinje slagati i nove, originalne rečenice. Ova analogija između ljudskog i umjetnog generiranja sadržaja otvara etička pitanja o slobodnoj volji i prirodi kreativnosti. Drugim riječima, postavlja se pitanje je li i čovjek jedna vrsta generatora sadržaja te je li naša kreativnost samo rezultat prethodnog iskustva koje smo prikupili tijekom života. To za sobom povlači i pitanje hoće li umjetna inteligencija biti toliko neovisna da će donositi samostalne odluke bez intervencije čovjeka te koliko bi to moglo biti opasno. Sve u svemu, generiranje umjetnog sadržaja predstavlja početak nove ere u umjetnoj inteligenciji, koja ne utječe samo na računarstvo, već zahvaća i mnoga druga društvena područja, otvarajući pritom brojna tehnička i etička pitanja.

U konačnici, ovaj rad je usmjeren na primjenu generativnih suparničkih mreža za generiranje medicinskih snimaka zajedno s pripadajućim maskama te će tema generiranja sadržaja biti razmatrana u tom pogledu. U ovom diplomskom radu, obrađena je tema medicinskih CT i MRI snimaka, zatim obrada medicinskih slika uz pomoć neuronskih mreža, problem nedostatka

medicinskih snimaka, što su to generativne suparničke mreže i kako ih primijeniti na generiranje medicinskih snimaka zajedno s pripadajućim maskama.

## **1.1. Zadatak diplomskog rada**

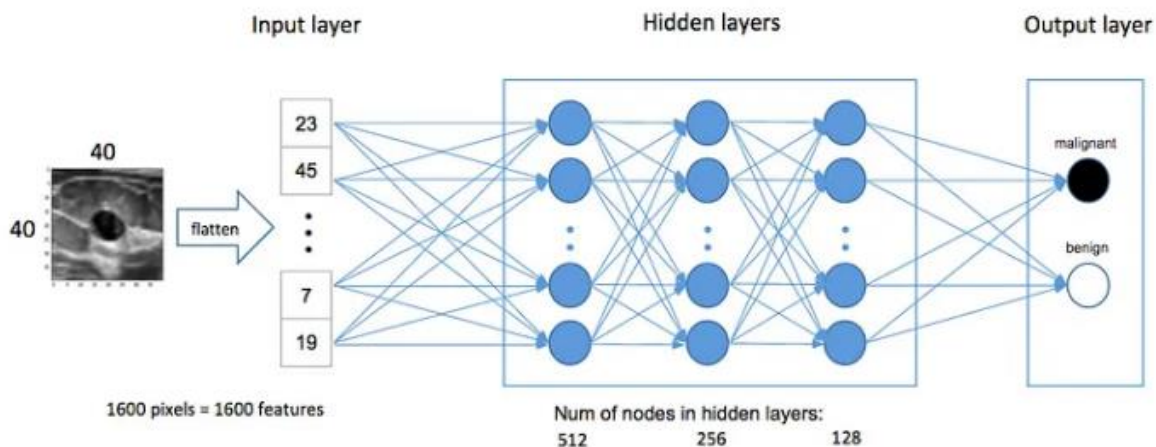
Zadatak diplomskog rada bio je istražiti i opisati način dobivanja medicinskih CT i MRI slika, objasniti teorijske osnove obrade medicinskih slika neuronskim mrežama, istražiti i opisati problematiku nedostatka medicinskih podataka za učenje neuronskih mreža, istražiti i opisati generativne suparničke mreže, razviti sustav za sintetičko generiranje originalnih i pripadnih anotiranih slika te prikazati i objasniti rezultate. Tehnologije koje se trebaju koristiti su Python, Keras te TensorFlow.



## 2. PREGLED PODRUČJA I PROBLEMATIKE

Najčešće korištene neuronske mreže u obradi medicinskih slika su konvolucijske neuronske mreže, U-Net te generativne suparničke mreže.

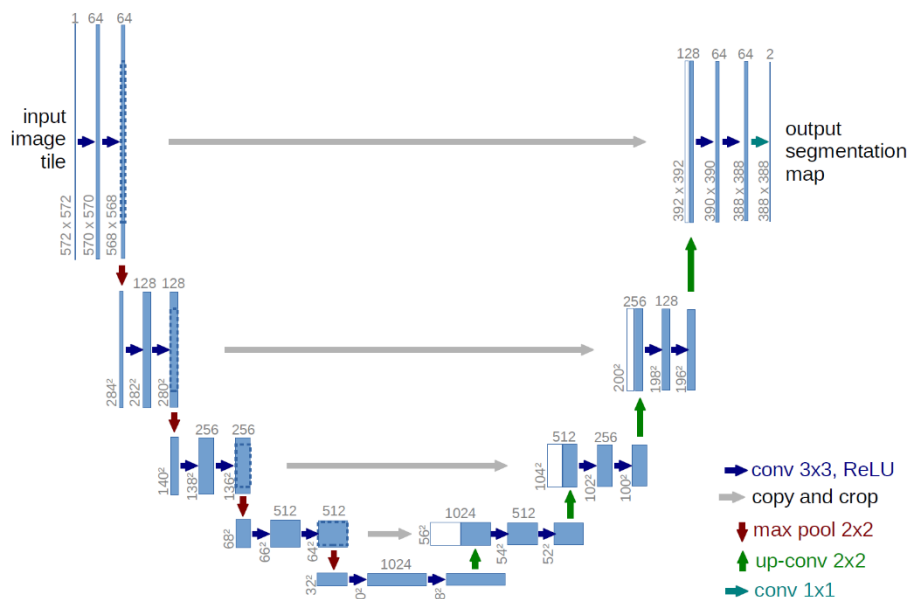
Konvolucijske neuronske mreže (*engl. Convolutional neural networks, CNN*) izrazito su učinkovite u prepoznavanju određenih objekata kroz konvolucijske slojeve. Konvolucija je matematička operacija koja stvara skup težina koji označava dio određene slike. Taj skup naziva se filter i manji je od ulazne slike. Filter se premješta kako bi se prikazao novi dio te se to ponavlja dok se čitava slika ne obradi. CNN-ovi su često primijenjena tehnika dubokog učenja za klasifikaciju bolesti, segmentaciju organa i lezija te detekciju anomalija. Na slici 2.1. prikazana je arhitektura konvolucijske neuronske mreže za automatsku detekciju karcinoma dojke na medicinskom snimku.



Slika 2.1. Arhitektura konvolucijske neuronske mreže za detekciju karcinoma dojke [1]

U-Net je vrsta konvolucijske neuronske mreže posebno razvijene za segmentaciju slika. Potrebno joj je manje slika za sam trening, a rezultati segmentacije su precizniji i točniji. Sastoji se od dva dijela: područja sažimanja ili enkodera te područja širenja ili dekodera. Sažimanje smanjuje

rezoluciju te izvlači ključne značajke, dok područje širenja ponovno rekonstruira sliku te se tako određeni objekti segmentiraju. Na slici 2.2. prikazana je arhitektura U-Net mreže.



Slika 2.2. U-Net arhitektura [2]

Generativne suparničke mreže (GAN) također su izrazito česta metoda korištena prilikom obrade medicinskih snimaka. GAN-ovi su najpoznatiji po generiranju umjetnog sadržaja, ali također služe i za poboljšavanje rezolucije te za prijenos određenih stilova i načina obrade s jedne slike na drugu. GAN mreže predstavljaju revolucionaran pristup u generiranju sintetičkih slika koje oponašaju stvarne podatke. GAN-ovi su osobito korisni u medicinskoj obradi slika zbog ograničenog broja dostupnih kvalitetnih i anotiranih podataka.

Jedna od najvažnijih primjena GAN-ova u medicinskoj obradi slika je sinteza visokokvalitetnih podataka koji se mogu koristiti za obuku modela u područjima kao što su klasifikacija, segmentacija i dijagnostika. Različite varijante GAN arhitektura pokazale su se učinkovitim u generiranju medicinskih slika. Primjerice, arhitektura DCGAN [3] (*engl. Deep Convolutional GAN*) bila je jedan od prvih pokušaja korištenja GAN-a s konvolucijskim slojevima. Iako su DCGAN-ovi jednostavni za implementaciju i stabilni tijekom treniranja, često imaju poteškoća s generiranjem visokokvalitetnih medicinskih slika zbog njihove sklonosti *mode collapse-u* što

rezultira ograničenim raznolikostima u generiranim slikama. Nadalje, StyleGAN [4] je jedan od najnaprednijih GAN-ova koji koristi progresivno treniranje i stilsku transformaciju latentnog prostora kako bi generirao visoko kvalitetne slike. U području medicinskih slika, StyleGAN je postigao impresivne rezultate, generirajući slike koje se teško razlikuju od stvarnih podataka. Ova arhitektura omogućava prilagodbu stila generiranih slika, što je korisno za različite modalitete u medicini, poput MRI, CT i fundusnih slika. SPADE GAN arhitektura [5] se posebno koristi za sintezu slika na temelju maski segmentacije. SPADE GAN omogućava preciznu kontrolu nad prostornom raspodjelom generiranih slika, što je korisno u medicinskim aplikacijama gdje je potrebno točno generirati slike specifičnih anatomskih struktura, poput srca ili jetre.

Iako GAN-ovi nude izvanredan potencijal za generiranje medicinskih slika, njihova primjena dolazi s određenim izazovima. Najznačajniji izazov je nedostatak podataka s obzirom na to da se u medicinskoj obradi slika često suočavamo s nedostatkom dostupnih i anotiranih podataka. Kreiranje medicinskih baza podataka je skupo i vremenski zahtjevno, što otežava treniranje GAN-ova na velikim skupovima podataka, kao što je slučaj i u drugim područjima računalnog vida.

Prema tome, najveći izazov u dubokom učenju je osigurati dovoljno kvalitetnih podataka za treniranje modela. Važno je da skup podataka bude vizualno kvalitetan, smislen, ali i dovoljno velik. Veći broj slika omogućava bolju preciznost i učinkovitost modela. DNN modeli bez velike količine kvalitetnih podataka ne mogu biti uspješni u generiranju novog sadržaja, detekciji i klasifikaciji bolesti niti uspješno prepoznavati obrasce.

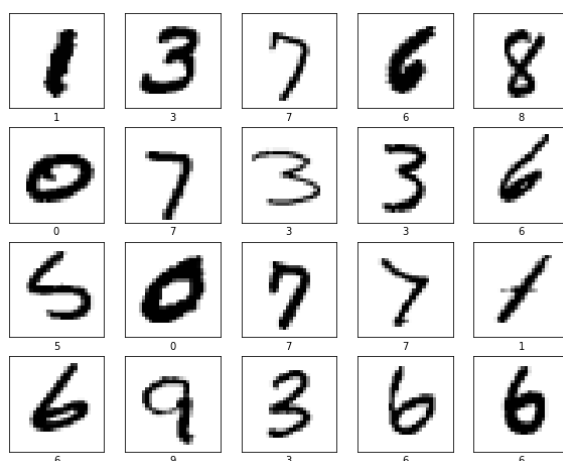
Nekoliko je razloga zašto nije jednostavno posjedovati podatkovni skup s dovoljno velikim brojem medicinskih snimaka. Osnovni je razlog privatnost pacijenata. Bolničke ustanove nerado dopuštaju uvid u privatne podatke pacijenata, uključujući i bolničke snimke. Nadalje, medicinske snimke nije jednostavno proizvesti jer one zahtijevaju specijaliziranu opremu i stručan kadar. Osim toga, snimke dolaze iz različitih izvora (npr. MRI, CT, rendgen), što znatno otežava njihovo korištenje jer ih je prvotno potrebno provesti kroz proces ujednačavanja formata. Najveći nedostatak snimaka je kod snimki određenih anomalija. Znatno je više dostupnih snimaka zdravih pacijenata, nego bolesnih, što otežava treniranje modela za prepoznavanje određenih bolesti.

Iako originalnih medicinskih snimaka nema u onom broju koji je potreban za uspješno treniranje DNN modela, moguće ih je kreirati ili nadopuniti na alternativni način. Prema [6], osim već

navedenog generiranja sintetičkih slika pomoću GAN-ova, teme koja je obrađena u nastavnim poglavljima ovog rada, nedovoljno velik skup podataka može se do određene mjere nadopuniti i augmentacijom podataka primjenom šumova, rotacije, skaliranja ili zrcaljenja na originalnom skupu podataka te transfornim učenjem. Primjenom tehnika za augmentaciju, podatkovni skup se povećava podacima koji su vrlo slični, no ipak sadrže određene male razlike, što je i ključno kod učenja modela. Transforno učenje vrlo je slično bibliotekama u programskim jezicima. Ono omogućuje korištenje velikih skupova podataka, koji su prethodno već bili istrenirani, na novim zadacima. Umjesto da se model trenira od nule, koriste se već istrenirani modeli koje se prilagođava na zadatke s manjim skupom podataka.

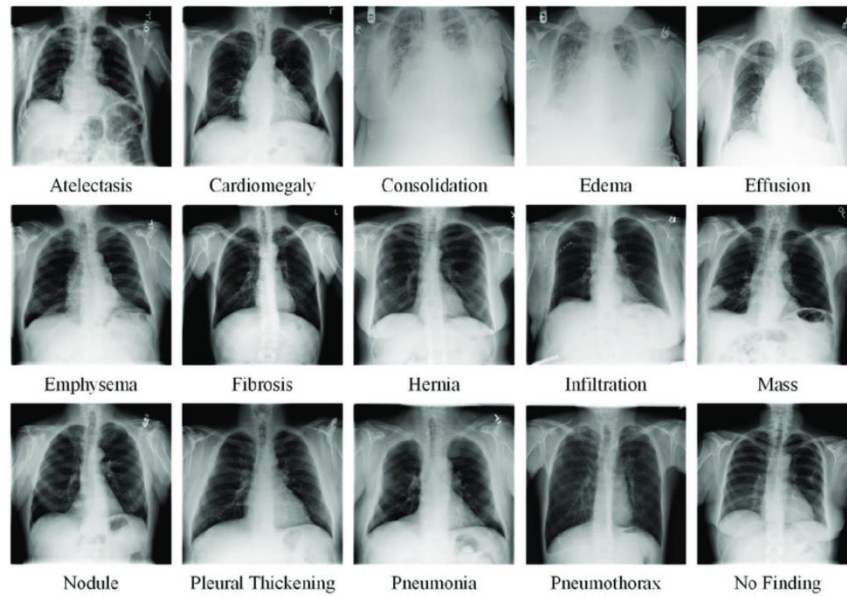
Primjeri često korištenih skupova podataka u transfornom učenju su:

- MNIST [7] (*engl. Modified National Institute of Standards and Technology database*) - Skup rukom pisanih znamenki. Vrlo jednostavan za obradu te se često koristi u početnim primjerima na temu dubokog učenja. Prikazan je na slici 2.3.



Slika 2.3. MNIST podatkovni skup [8]

- ImageNet [9] - Jedan od najvećih i najčešće korištenih skupova podataka. Slike su razdvojene u razne kategorije.
- ChestX-ray14 [10] - Skup podataka s više od sto tisuća snimaka prsnog koša s primjerima različitih bolesti. Prikazan je na slici 2.4.

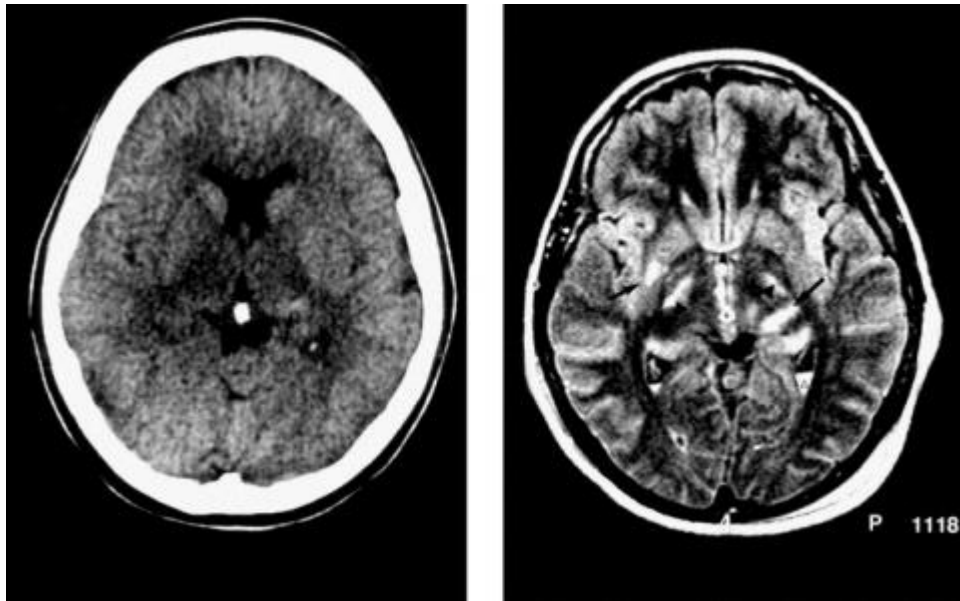


Slika 2.4. ChestX-ray14 podatkovni skup [11]

- LUNA16 [12] (*engl. Lung Nodule Analysis*) - Skup podataka s više od tisuću CT snimaka pluća.

### 3. MEDICINSKE CT I MRI SNIMKE

Računalna tomografija i magnetska rezonancija su metode prikupljanja snimaka unutarnjih struktura ljudskog tijela. Navedene dijagnostičke metode omogućuju detaljan prikaz određenih dijelova tijela što stručnjacima uvelike olakšava daljnju analizu, otkrivanje određenih anomalija te praćenje samog tijeka liječenja. Slika 3.1. prikazuje CT i MRI snimku mozga.



Slika 3.1. CT snimka mozga (lijevo) i MRI snimka mozga (desno) [13]

#### 3.1. Računalna tomografija

Računalna tomografija (*engl. computed tomography, CT*) je radiološka metoda koja pomoću ionizirajućeg rendgenskog zračenja slojevito prikazuje unutarnje strukture tijela. Funkcionira tako da se rendgenski izvor i detektor rotiraju oko pacijenta, pri čemu senzori detektora bilježe rendgenske zrake koje prolaze kroz tijelo. Prikupljeni podatci se potom obrađuju računalom kako bi se stvorile dvodimenzionalne ili trodimenzionalne slike organa i tkiva.

CT može otkriti poremećaje u gotovo svim dijelovima tijela. Primjerice, može identificirati i izmjeriti veličinu tumora, prepoznati prijelome kostiju, otkriti upalu pluća, prirodene malformacije te moždana krvarenja.

Međutim, nedostatak CT snimanja je u upotrebi značajno veće doze zračenja u usporedbi s klasičnim rendgenom. Prema [14], snimanje abdomena može koristiti 300 do 400 puta više zračenja nego standardni rendgen prsnog koša. CT snimanja odgovorna su za najveći udio zračenja uzrokovanog medicinskim dijagnostičkim postupcima u općoj populaciji te se ne preporučuje djeci i trudnicama. Slika 3.2. prikazuje CT snimku trbušne šupljine i zdjelice.



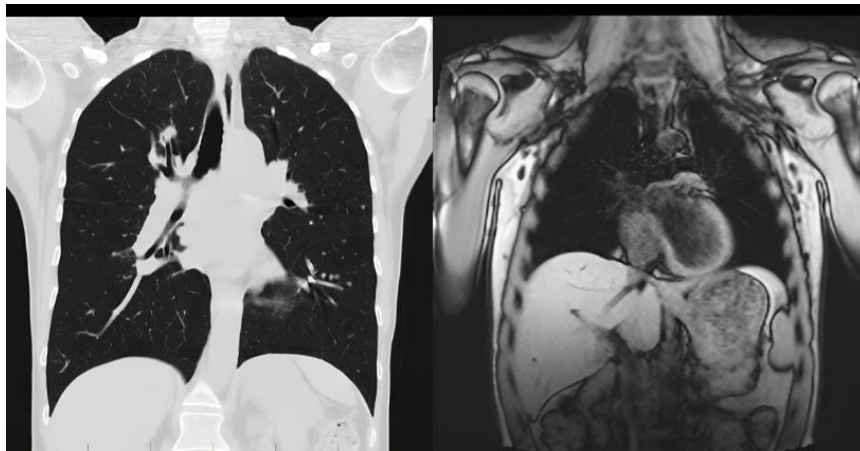
Slika 3.2. CT snimka trbušne šupljine i zdjelice [15]

### 3.2. Magnetska rezonancija

Magnetska rezonancija (*engl. magnetic resonance, MR*) koristi magnetsko polje i radio valove visoke frekvencije kako bi dobila slikovne prikaze unutarnjih struktura tijela koji se nazivaju MRI (*engl. magnetic resonance imaging*). Za razliku od CT-a, ne koristi se rendgenskim zrakama.

MR funkcionira tako da se protoni u MR skeneru poredaju prema jakom magnetskom polju. Skener zatim emitira puls radio valova koji na trenutak izbacuje protone iz njihovog usmjerenja. Prilikom vraćanja protona u prvotni poredak u smjeru s magnetskim poljem, oni oslobode energiju, to jest signale koje MR skener zatim snimi i na temelju čega se izrade snimke. Promjenom pulsa radio valova, snage i smjera magnetskog polja, može se promijeniti način na koji se tkiva prikazuju. Primjerice, masno tkivo može biti tamno na jednoj snimci, a svijetlo na drugoj. Kako bi slike bile jasnije, u venu ili zglob se može ubrizgati kontrastno sredstvo koje sadrži gadolinij.

MR-u se daje prednost nad CT-om kada je u pitanju snimanje mekih tkiva, primjerice kada je potrebno snimiti mozak, kralježničnu moždinu ili mišiće [16]. Zbog detaljnog prikaza organa i tkiva, MR se pokazao kao izrazito koristan u identificiranju tumora. Nadalje, budući da je MR sigurniji od CT-a, primjenjuje se i u slučaju kad su rizici od oštećenja izlaganjem CT snimanju previsoki. S druge strane, vrijeme potrebno za izradu MRI snimki duže je nego ono potrebno za CT. MR je skuplji i teže dostupan. Osim toga, ne smije se primjenjivati za snimanje osoba koje imaju metalne uređaje ugrađene u tijelo. Na slici 3.3. prikazani su CT, a desno MRI snimke mozga i prsa.



Slika 3.3. CT snimka prsa (lijevo) i MRI snimka prsa (desno) [4]



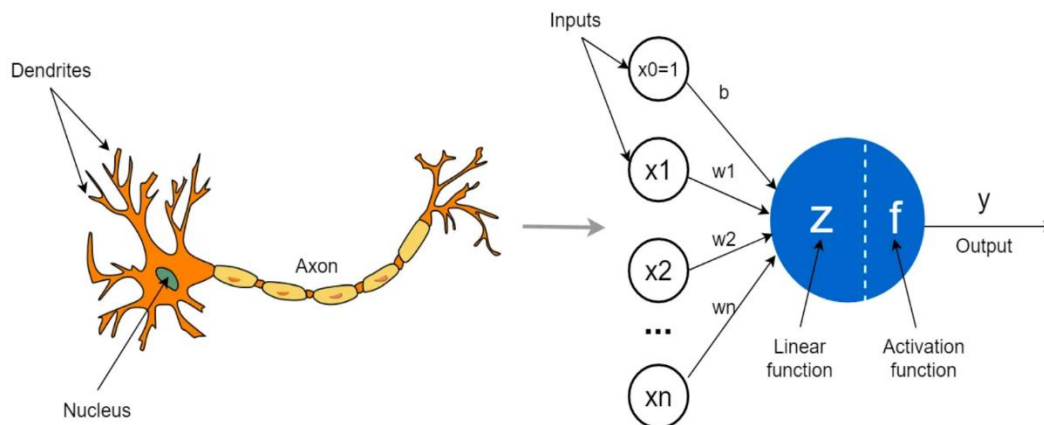
## 4. OBRADA MEDICINSKIH SLIKA NEURONSKIM MREŽAMA

Primjena neuronskih mreža u obradi medicinskih slika značajno je unaprijedila dijagnostiku i analizu različitih bolesti. U nastavku su objašnjene neuronske mreže te istaknuti ključni primjeri njihove primjene, uključujući klasifikaciju, segmentaciju, detekciju objekata te generiranje podataka.

### 4.1. Neuronska mreža

Neuronska mreža (*engl. artificial neural network, ANN ili neural net, NN*) model je strojnog učenja nastao na temelju strukture i funkcije biološke neuronske mreže u mozgu.

Osnovna jedinica neuronskih mreža su spojeni čvorovi koji se nazivaju umjetni neuroni. Oni imitiraju funkciju neurona u mozgu te međusobno komuniciraju umjetnim vezama kao što neuroni u mozgu komuniciraju sinapsama. Sinapse u biološkom mozgu označavaju neuronske puteve koji se formiraju svaki put kada čovjek uči. Prema [17], stjecanjem novog ili osvježavanjem već stečenog znanja, veza između neurona jača te se komunikacija između njih poboljšava. Slično tome, ponavljanje u neuronskim mrežama jača veze između umjetnih neurona što poboljšava prijenos signala između njih. Slika 4.1. prikazuje usporedbu biološkog i umjetnog neurona.

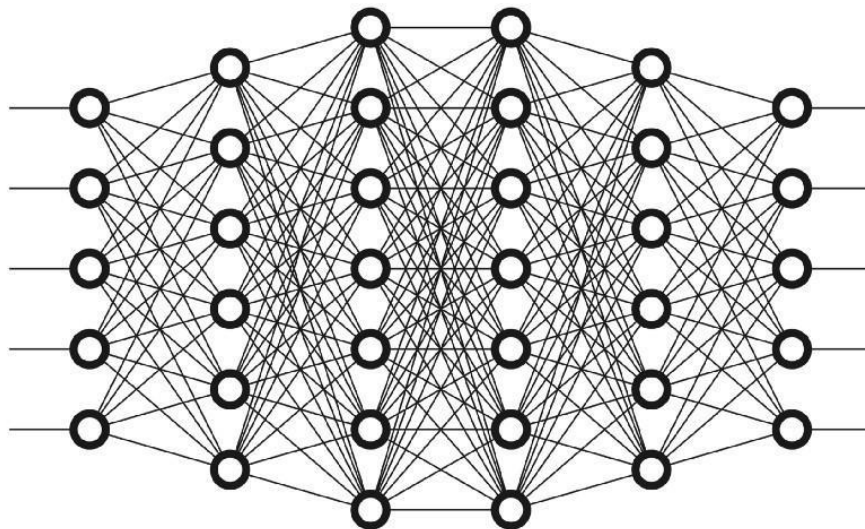


Slika 4.1. Biološki neuron (lijevo) i umjetni neuron (desno) [18]

Signal se prenosi tako da ga umjetni neuron primi od povezanog neurona, obradi ga te ga prosljedi sljedećem povezanom neuronu. Na temelju ulaznog signala i njegove težine, aktivacijska funkcija izračuna izlaz za svaki umjetni neuron. Težinom je određena jačina svakog signala te se njena vrijednost prilagođava tijekom procesa učenja.

#### 4.1.1. Duboka neuronska mreža

Neuroni su obično organizirani u slojeve koji mogu obavljati različite transformacije podataka. Signali putuju od ulaznog do izlaznog sloja, prolazeći kroz više skrivenih slojeva. Mreža koja sadrži najmanje dva skrivena sloja, naziva se dubokom neuronskom mrežom (*engl. deep neural network, DNN*). DNN-ovi se koriste u dubokom učenju za obradu velikih količina podataka. Njihova primjena je raznolika, od računalnog vida i prepoznavanja govora do analize medicinskih slika i bioinformatike. Na slici 4.2. prikazan je koncept duboke neuronske mreže.



Slika 4.2. Koncept duboke neuronske mreže: ulazni sloj, dva ili više skrivenih slojeva te izlazni sloj [19]

## 4.2. Primjena neuronskih mreža u obradi medicinskih slika

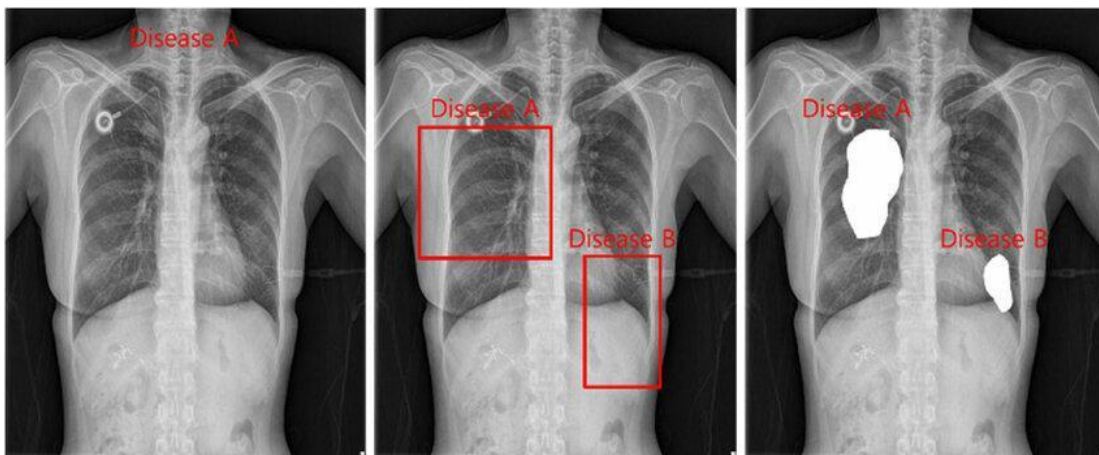
Duboke neuronske mreže primjenjuju se u različitim područjima obrade medicinskih slika, uključujući klasifikaciju, detekciju bolesti te segmentaciju. Njihova sposobnost izdvajanja značajki i obrazaca iz složenih medicinskih snimki čini ih iznimno korisnima u dijagnostici i analizi snimki. Korištenje DNN-ova olakšava prepoznavanje anomalija, poboljšava preciznost dijagnoza i omogućuje preciznije prognoziranje ishoda liječenja.

Računalno potpomognuti dijagnostički sustavi (*engl. Computer-Aided Diagnosis, CAD*) značajno su poboljšani integracijom DNN-ova u sustav. Starije verzije CAD-a oslanjale su se na ručno definirane značajke i strogo strukturirane algoritme, što je ograničavalo njihovu preciznost. Uvođenjem DNN-ova, moderni CAD sustavi postali su daleko točniji i učinkovitiji u analizi medicinskih slika.

U nastavku su navedeni najčešći primjeri primjene DNN-ova u obradi medicinskih snimaka.

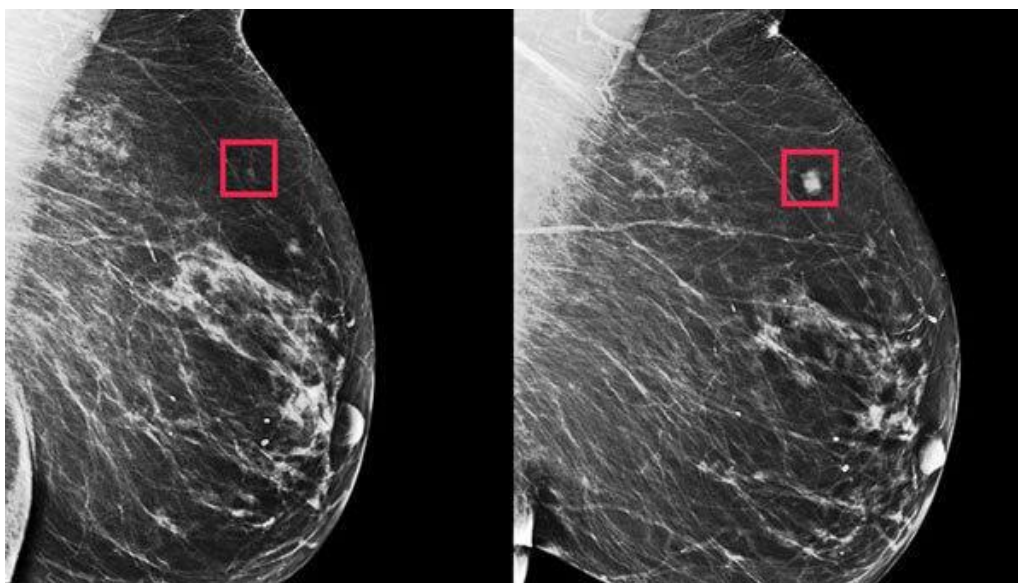
### 4.2.1. Klasifikacija, segmentacija i detekcija objekta

Trenirajući na velikim anotiranim skupovima podataka, DNN-ovi mogu detektirati anatomske strukture te ih klasificirati ili segmentirati (Slika 4.3.). Klasifikacija omogućuje prepoznavanje tipa ili klase objekta, segmentacija precizno definira njegov obris, dok detekcija identificira objekt unutar slike.



Slika 4.3. Prva snimka prikazuje klasifikaciju, druga detekciju, a treća segmentaciju objekta [20]

Upravo primjenjujući ove tehnike, DNN-ovi mogu uspješno otkriti i identificirati anomalije. Prema [21], na temelju velikih skupova medicinskih slika, mogu se identificirati biomarkeri povezani s određenim bolestima. Primjerice, na snimkama raka moguće je prepoznati ili predvidjeti malignost tumora. Na slici 4.4. prikazane su mamografske snimke pacijentice kod koje je algoritam prepoznao visoku vjerojatnost razvoja karcinoma dojke čak četiri godine prije nego što je bolest uistinu i dijagnosticirana.



Slika 4.4. Detekcija značajke koja ukazuje na visoku mogućnost razvitka karcinoma (lijevo) te dijagnosticiran karcinom dojke četiri godine kasnije (desno) [22]

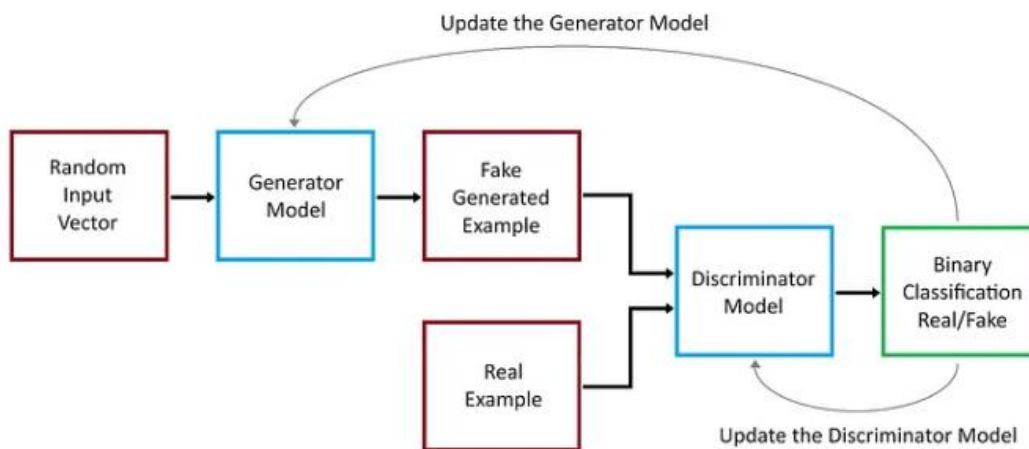
#### 4.2.2. Rekreiranje, registracija te generiranje slika

Kvaliteta medicinskih snimaka može se značajno poboljšati dubokim neuronskim mrežama. CT ili MRI snimke koje sadrže veliku količinu šuma vrlo se efikasno poboljšavaju primjenom algoritama dubokog učenja. Nadalje, DNN-ovi se koriste za registraciju slika, odnosno za usklađivanje snimaka dobivenih iz različitih modaliteta ili vremenskih razdoblja. Korištenjem prostornih transformacija, DNN-ovi omogućuju precizno poravnavanje i usporedbu slika. Još jedan važan aspekt DNN-ova je njihova sposobnost generiranja medicinskih snimki, što je i ključna tema ovog rada. Generiranje novih snimaka je izrazito važno ako se radi o rijetkim

bolestima. U tom slučaju generirane slike omogućuju dovoljno velik podatkovni skup kako bi se model mogao istrenirati te u konačnici naučiti kako identificirati određenu bolest.

### 4.3. Generativne suparničke mreže

Generativne suparničke mreže (*engl. Generative Adversarial Networks, GAN*) napredni su modeli nenadziranog dubokog učenja. Osnovna karakteristika GAN-ova su dvije mreže koje se sukobljavaju: generator, koji generira podatke, i diskriminator, koji ima zadatak razlikovati stvarne podatke od onih koji su generirani. Generator prima vektor slučajnih vrijednosti kao ulaz i transformira ga u lažne, to jest generirane podatke. Oni se zajedno sa stvarnim podacima predaju diskriminatoru koji pokušava odrediti koji podatci su realni, a koji generirani. Ako generator uspije prevariti diskriminator, znači da diskriminator treba unaprijediti. Ako diskriminator prepozna podatke generatora kao umjetno generirane, znači da generator nije dovoljno dobar te se mora poboljšati. Takav ciklus se ponavlja te s vremenom rezultira sve boljim rezultatima, to jest umjetno generiranim slikama visoke kvalitete koje izgledaju realno. Na slici 4.5. prikazan je model generativne suparničke mreže.



Slika 4.5. Model generativne suparničke mreže [23]

### 4.3.1. Nastanak i razvoj generativnih suparničkih mreža

Generativne suparničke mreže je 2014. predstavio Ian Goodfellow zajedno sa svojim suradnicima. Prema Goodfellowu [24], on je htio razviti model koji bi omogućio generiranje podataka koji su slični onima u postojećem skupu te je odlučio sukobiti dvije neuronske mreže, generatora i diskriminatora, jednu s drugom. Ova koncepcija ubrzo je postala popularna jer je omogućila generiranje realističnih podataka iz neoznačenih primjera.

Budući da osnovna verzija GAN-a ima određene nesavršenosti poput nestabilnosti treniranja te *mode collapse-a* (generiranje vrlo sličnih podataka), u međuvremenu su se razvile razne varijante GAN-ova koje su nudile rješenja za gore navedene poteškoće. 2015. se pojavljuju DCGAN-ovi (*engl. Deep Convolutional GAN*) koji dodatnom primjenom konvolucijskih neuronskih mreža generiraju kvalitetnije slike od samih GAN-ova. Nedugo nakon toga, 2017. razvijen je WGAN (*engl. Wasserstein GAN*) koji zamjenjuje binarnu unakrsnu entropiju (*engl. binary cross entropy*) s Wasserstein gubitkom [25]. Ova promjena omogućila je stabilnije treniranje i smanjila pojavu *mode collapse-a*.

GAN-ovi su se kasnije razvili na razne načine. Među najpoznatijim specijaliziranim verzijama je već spomenuti StyleGAN, koji je sposoban kreirati izrazito realistična ljudska lica, te CycleGAN, koji je postao poznat po stilizacijama slika te transformacijama između različitih domena, primjerice prikaz zebrostog uzorka na koži lava.

### 4.3.2. Izazovi i poteškoće prilikom rada s GAN-ovima

Iako su GAN-ovi revolucionarni modeli koji su unaprijedili strojno učenje, postoje određeni problemi koji otežavaju rad na njima [26]. Najučestalije poteškoće s kojima se stručnjaci susreću prilikom kreiranja GAN-ova su:

- nestabilno treniranje
- *mode collapse* te *mode dropping*
- osjetljivost na parametre
- nedostatak mjerne funkcije

- izrazito velika potrošnja resursa.

Osnovni izazov prilikom rada na GAN modelima je nestabilno treniranje. Budući da se GAN temelji na sukobu generatora i diskriminatora, može doći do situacije u kojoj treniranje oscilira. Diskriminator može nadjačati generatora ili treniranje može postati nekontrolirano ako su primjerice gradijenti preveliki. Nadalje, *mode collapse*, to jest generiranje vrlo sličnog sadržaja, je najčešći problem kojeg je izrazito teško izbjeći ako se model temelji na osnovnoj verziji GAN-a koja primjenjuje binarnu unakrsnu entropiju. Osim *mode collapse-a*, još jedan problem je i *mode dropping*, kada generator zanemari određene dijelove podataka te generira samo dio slike, umjesto cijeli kontekst. Nadalje, GAN-ovi su osjetljivi na parametre te pronalaženje kombinacije koja daje dobre rezultate može biti poprilično iscrpljujuće. Nedostatak mjerne funkcije je također jedan od problema, budući da je kvalitetu GAN-a moguće provjeriti samo usporedbom sa stvarnim rezultatima, što nije dovoljno egzaktna ni točna metoda. Izrazito velika poteškoća je i velika potrošnja resursa. GAN-ovi su vremenski i računalno vrlo zahtjevni. Treniranje prosječnog podatkovnog skupa traje satima te iziskuje kvalitetni grafički procesor.

### **4.3.3. Treniranje generativnih suparničkih mreža**

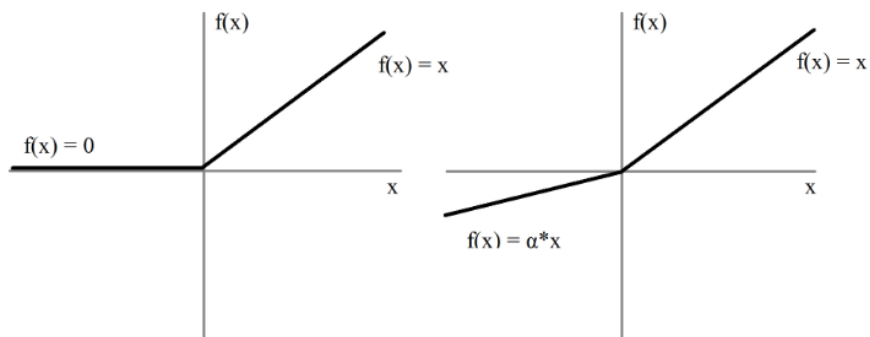
Treniranje GAN-ova temelji se na repetitivnom sukobljavanju generatora i diskriminatora. Budući da su svi ulazni podatci za GAN neoznačeni, treniranje GAN-a svrstava se u metode nenadziranog učenja.

#### **4.3.3.1. Arhitektura generatora**

Generator ima dva ulaza: ulaz za slučajni šum te ulaz za podatke. Na temelju tih ulaza, model generira podatke koji nalikuju onim ulaznim. Na početku generira nasumične šumove, no kako se GAN trenira, generator uči proizvoditi podatke sve sličnije stvarnoj distribuciji. Tijekom treniranja, generator prilagođava svoje težine na temelju povratnih informacija od diskriminatora. Ako diskriminator ispravno prepozna lažne podatke, generator prilagođava svoje težine kako bi sljedeći put proizveo uvjerljivije podatke.

Generator se sastoji od ulaznog sloja koji prima vektor slučajnog šuma, potpuno povezanih slojeva koji transformiraju ulazni šum u oblik prikladan za daljnju obradu, batch normalizacije, tehnike koja stabilizira učenje normalizacijom izlaza prethodnih slojeva, aktivacijske funkcije, uglavnom ReLU ili Leaky ReLU, koja omogućuje stvaranje složenih podataka, transponiranih konvolucijskih slojeva koji povećavaju prostornu dimenziju ulaza, slojeva za promjenu oblika, koji pretvaraju podatke u željeni izlazni format, te izlaznog sloja, koji je završni sloj koji pomoću tanh ili sigmoid funkcije normalizira vrijednost piksela.

ReLU (*engl. Rectified Linear Unit*) i Leaky ReLU su dvije slične funkcije koje uvode nelinearnost u mrežu prilikom dubokog učenja te time omogućuju učenje složenih obrazaca. Problem kod ReLU je da može doći do neaktivnih neurona jer su njihovi izlazi konstantno 0 za sve ulaze manje ili jednake od 0. Leaky ReLU rješava problem neaktivnih neurona jer se negativne vrijednosti ne gube, već se skaliraju [27]. Na slici 4.6. prikazane su funkcije ReLU i Leaky ReLU.



Slika 4.6. ReLU funkcija (lijevo) te Leaky ReLU funkcija (desno) [28]

#### 4.3.3.2. Arhitektura diskriminatora

Diskriminator ima zadatak razlikovati stvarne podatke od generiranih. Na ulazu prima podatke, a na izlazu daje skalarnu vrijednost između 0 i 1, koja označava vjerojatnost da je ulaz stvaran. Vrijednost bliža 1 označava stvaran podatak, dok vrijednost bliža 0 sugerira da je podatak generiran.



Diskriminator je po arhitekturi sličan konvolucijskim neuronskim mrežama. Sastoji se od konvolucijskih slojeva koji omogućuju ekstrakciju značajki iz slikovnih podataka, batch normalizacije koja stabilizira treniranje, i aktivacijskih funkcija. Neke arhitekture koriste *pooling* slojeve, poput *max pooling*-a, za smanjenje prostornih dimenzija. Na kraju mreže, potpuno povezani slojevi obrađuju značajke i vode do završnog sloja, gdje se dobije vjerojatnost da je ulazna slika stvarna ili generirana.

#### 4.3.3.3. Funkcija gubitka

Generativne suparničke mreže koriste funkcije gubitka kako bi optimizirale težine generatora i diskriminatora tijekom treniranja, čime poboljšavaju ukupnu izvedbu modela. Funkcija gubitka koja se koristi u osnovnoj verziji GAN-a je binarna unakrsna entropija. Ona mjeri razliku između stvarnih i predviđenih binarnih oznaka, pomažući modelima da preciznije razlikuju stvarne podatke od generiranih [29]. Formula 4-1 predstavlja binarnu unakrsnu entropiju (BCE):

$$BCE = -\frac{1}{N} \sum_{i=1}^N y_i * \log \log (p(y_i)) + (1 - y_i) * \log(1 - p(y_i)) \quad (4-1)$$

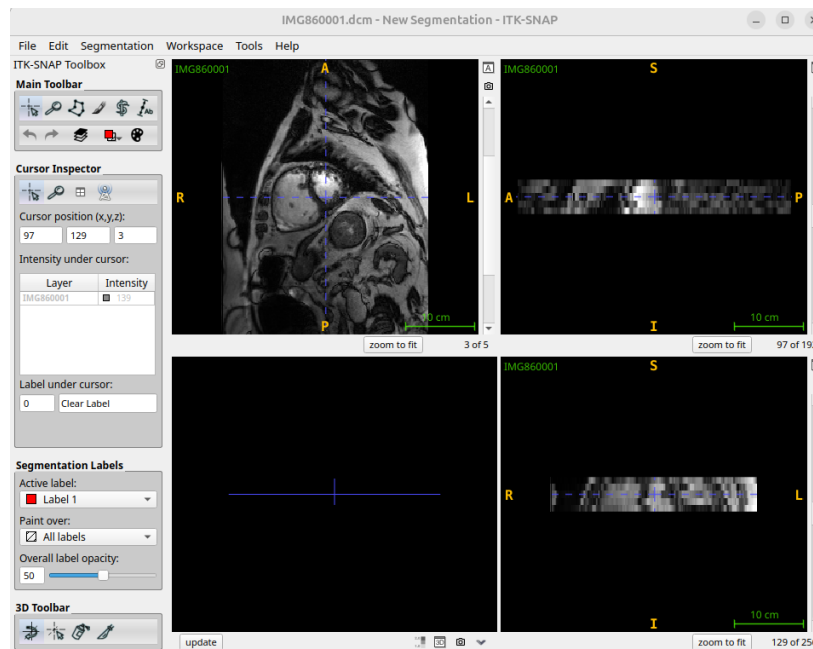
pri čemu je  $p(y_i)$  vjerojatnost vrijednosti 1, a  $1 - p(y_i)$  vjerojatnost vrijednosti 0.

## 5. MODEL GAN-A ZA GENERIRANJE MEDICINSKIH SNIMAKA I PRIPADNIH ANOTIRANIH MASKI

U ovom dijelu rada detaljno je prikazan i objašnjen kreirani model za generiranje medicinskih snimaka i njima pripadajućih anotiranih maski primjenom generativnih suparničkih mreža.

### 5.1. Podatkovni skup

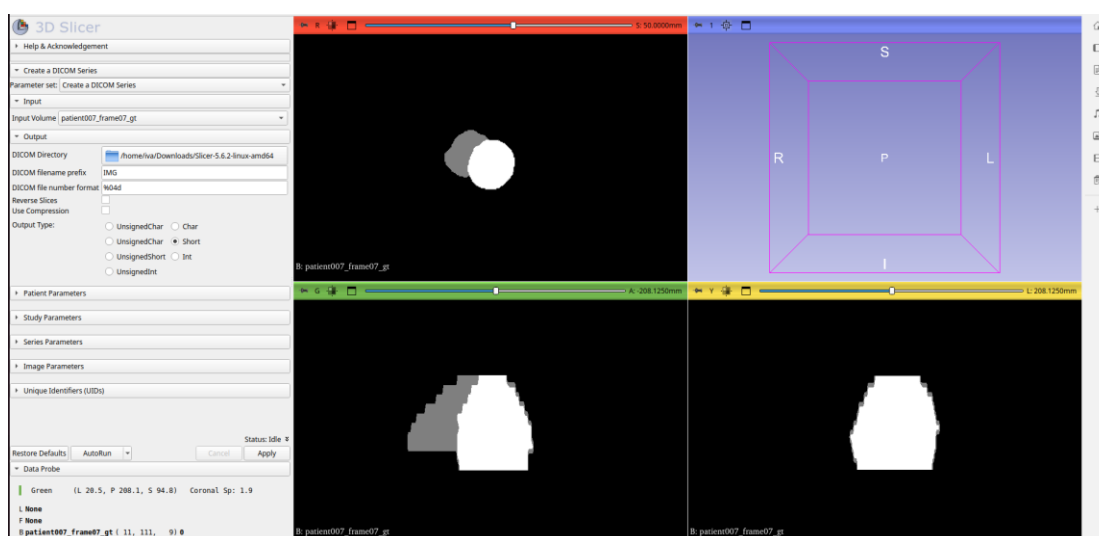
Podatkovni skup sastojao se od medicinskih snimki te pripadajućih maski. Slike iz podatkovnog skupa bile su u 3D NIfTI formatu te su se mogle pregledati pomoću aplikacije ITK-Snap. Na slici 5.1. prikazano je sučelje aplikacije s učitanim snimkom iz podatkovnog skupa.



Slika 5.1. Aplikacija ITK-Snap

NIFTI (*engl. Neuroimaging Informatics Technology Initiative*) format omogućuje spremanje medicinskih snimaka u 3D formatu, zajedno s njihovim opisima, te je kompatibilan s većinom programskih rješenja i alata.

Budući da su slike u 3D formatu, bilo ih je potrebno pretvoriti u 2D kako bi se mogle koristiti kao podatkovni skup za treniranje. Prebačene su u 2D format pomoću 3D Slicer-a. Sučelje aplikacije s učitanim maskom iz podatkovnog skupa vidljivo je na slici 5.2. To je programska podrška koja se koristi za obradu slika iz područja medicine, biomedicine i slično. Snimke i pripadajuće maske su pojedinačno učitane u program te je u prosjeku od jedne 3D slike, to jest maske, dobiveno 10 2D slika iliti *slice*-eva.

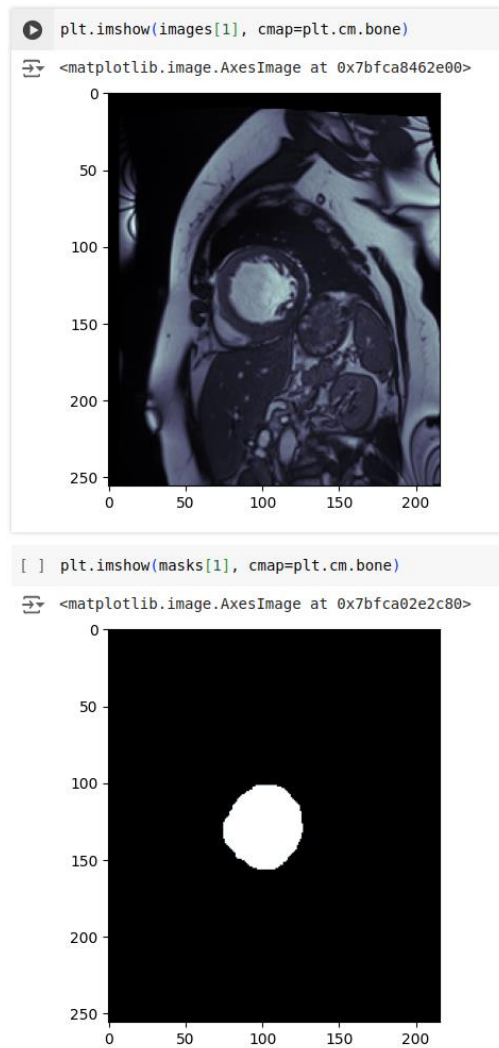


Slika 5.2. 3D Slicer

Nadalje, *slice*-evi kreirani u programu prebačeni su u DICOM format pomoću opcije *Create DICOM series*. DICOM (*engl. Dicom Imaging and Communications in Medicine*) je standardizirani format za pohranu medicinskih slika te za njihovu daljnju obradu u dubokom učenju.

Nakon što su prebačene u 2D format, slike i maske su manualno razdvojene u dvije različite datoteke. Svaka snimka i njoj pripadajuća maska bile su povezane sufiksom imena u obliku broja.

Primjerice, snimci IMG00001 pripadajuća maska u susjednoj datoteci zove se IMG00001. Na slici 5.3. vidljiv je primjer snimke iz podatkovnog skupa i njoj pripadajuće maske. Nadalje, budući da je kvalitetan podatkovni skup nužan za uspješno generiranje sadržaja, pomoću aplikacije Snap-ITK pregledane su snimke u podatkovnom skupu te su eliminirane one koje izrazito odstupaju od prosjeka.



Slika 5.3. Prikaz medicinske snimke iz podatkovnog skupa i pripadajuće maske

## 5.2. Razvojno okruženje i korištene biblioteke

U nastavku je navedeno razvojno okruženje korišteno za kreiranje modela te su istaknute najvažnije biblioteke.

### 5.2.1. Razvojno okruženje

Programsko rješenje pisano je u programskom jeziku Python u Google Colaboratory-u (Google Colab). Google Colab je razvojno okruženje koje omogućuje pokretanje Jupyter Notebooka u oblaku te korištenje računalnih resursa poput grafičkih procesora (GPU) ili tenzorskih procesora (TPU).

Generirati nove podatke koristeći GAN-ove vrlo je iscrpno te zahtjeva korištenje značajnih resursa. GAN-ovi sadrže kompleksne matematičke izračune, uglavnom su trenirani na velikim skupovima podataka te njihovo treniranje izrazito dugo traje. Kako bi takav proces bio brži i efikasniji, potrebno je koristiti kvalitetan grafički procesor (GPU) koji može značajno ubrzati treniranje modela, omogućujući rad na kompleksnijim modelima i većim skupovima podataka. Ova ubrzanja su presudna za postizanje kvalitetnih rezultata. Prilikom treniranja ovog konkretnog modela, korišten je GPU A100 koji je ponuđen u Google Colab Pro verziji.

Colab u sebi sadrži predefimirane biblioteke koje su primjenjive na zadatke vezane uz strojno učenje, posebice duboko učenje.

### 5.2.2. Ključne biblioteke

Biblioteke su od izrazite važnosti prilikom programiranja, posebno kada se radi o kompleksnim algoritmima koji se koriste prilikom kreiranja modela dubokog učenja. One smanjuju potrebu za ponovnim pisanjem koda, smanjuju mogućnost pogreške te ubrzavaju dolazak do rezultata.

Neke od važnijih datoteka korištenih u pisanju ovog modela su *NumPy*, biblioteka koja omogućava rad s višedimenzionalnim nizovima i matricama, *pydicom* koji se koristi za rad sa slikama u DICOM formatu, *TensorFlow* koji omogućava izgradnju i treniranje neuronskih mreža, te *Keras* koji je zapravo API TensorFlowa te pokriva sve faze razvoja modela strojnog učenja, od obrade podataka do implementacije samog modela.

### 5.3. Pretvaranje snimaka i maski u NumPy nizove i normalizacija maski

Nakon pozivanja važnih biblioteka, učitavaju se medicinske snimke i pripadajuće maske koje su u DICOM formatu. Funkcija `os.listdir` dohvaća popis datoteka u direktoriju `images`, to jest `masks`. Zatim se pomoću funkcije `zip` snimke i maske upare kako bi mogle biti istovremeno obrađene. Nadalje, snimke se učitavaju funkcijom `pydicom.dcmread`, pretvaraju se u niz piksela funkcijom `pixel_array` te ujednačavaju se na istu veličinu funkcijom `cv2.resize`. Isti postupak ponavlja se i za maske. Zatim se snimke pohranjuju u listu `images`, maske u listu `masks`, te se konačno pretvaraju u NumPy nizove pomoću funkcije `np.array`. Nakon toga, budući da maske nisu toliko kompleksne kao snimke, vrši se normalizacija na vrijednosti između 0 i 1 te binarizacija u smislu da su sve vrijednosti iznad 0.5 1, to jest bijele, a sve ispod 0.5 0, to jest crne. Programski kod prikazan je na slici 5.4.

```
images = []
masks = []
for image, mask in zip(os.listdir('images'), os.listdir('masks')):
    i = pydicom.dcmread('images/' + image)
    i = i.pixel_array
    i = cv2.resize(i, dsize = (216, 256))
    m = pydicom.dcmread('masks/' + mask)
    m = m.pixel_array
    m = cv2.resize(m, dsize = (216, 256))
    images.append(i)
    masks.append(m)

images = np.array(images)
masks = np.array(masks)
masks = masks.astype(np.float32)
masks = masks / np.max(masks)
masks = (masks >= 0.5).astype(np.float32)
```

Slika 5.4. Pretvaranje slika u NumPy nizove te normalizacija maski

## 5.4. Predprocesiranje podataka

Prilikom korištenja generativnih suparničkih mreža, važno je podatke pripremiti za treniranje modela. Priprema podataka na treniranje naziva se predprocesiranje podataka.

Pomoću funkcije *pick\_large\_contours* filtriraju se snimke i pripadajuće maske prema veličini kontura. Izabiru se one snimke i maske koje imaju dovoljno velike konture objekata. Nadalje, radi lakšeg generiranja, visina i širina se postavljaju na 128 piksela, a normalizacija se provodi *minusplusone* načinom, to jest da je vrijednost piksela između -1 i 1. Pomoću funkcije *concatenate*, snimke i maske se zajedno uparuju kako bi se mogle koristiti kao ulazni podatak u daljnjem procesu treniranja. Programski kod prikazan je na slici 5.5., a obrađene slike snimaka i maski na slici 5.6.

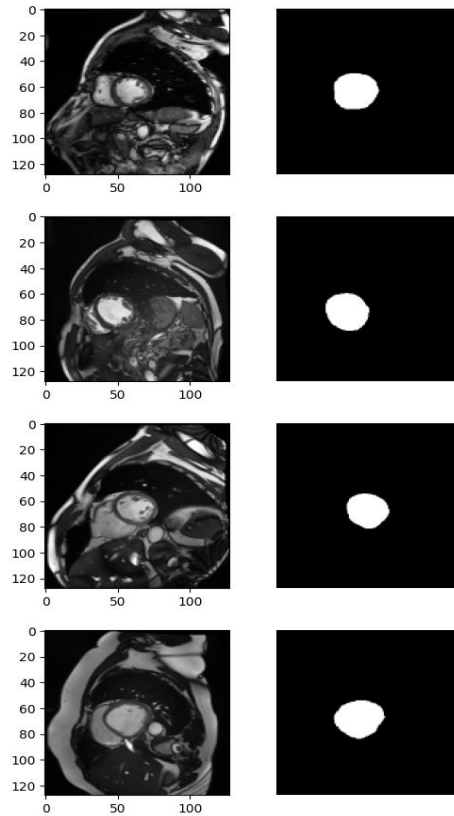
```
X_train = np.load('images.npy')
Y_train = np.load('masks.npy')
get_stats(X_train)
get_stats(Y_train)

X_train,Y_train = pick_large_contours(X_train,Y_train,2000)
get_stats(X_train)
get_stats(Y_train)

params={
    'h': 128,
    'w': 128,
    'norm_type': 'minusplusone',
}
X_train,Y_train = preprocess(X_train,Y_train,params)
get_stats(X_train)
get_stats(Y_train)

X_train = np.concatenate((X_train,Y_train),axis=3)
get_stats(X_train)
```

Slika 5.5. Predprocesiranje podataka



Slika 5.6. Obradene slike snimaka i maski

## 5.5. Generator

Generator je, uz diskriminator, temeljni dio GAN modela. Ovdje je definiran koristeći biblioteke *TensorFlow* i *Keras*. Prvotno se definira *Dense* sloj koji uzima nasumični šum vektora dimenzije 100 te inicijalizira težine u modelu putem normalne distribucije. Nakon toga, funkcijom *BatchNormalization*, izlaz iz prethodnog sloja se stabilizira kako bi se ubrzalo treniranje. Zatim se pomoću funkcije *Reshape* vektor preoblikuje u trodimenzionalni tenzor kojem je treća dimenzija broj kanala. Nakon toga dolazi do povećanja prostorne dimenzije podataka pomoću interpolacije te se dodaje konvolucijski sloj kako bi se slika proširila s dodatnim značajkama. Zatim opet dolazi do normalizacije, ali s manjim momentumom nego u početnim slojevima. Aktivacijska funkcija je *Leaky ReLU* koja stabilizira treniranje. Ona omogućuje malim negativnim vrijednostima da prođu kroz funkciju. Tako, čak i kad su vrijednosti negativne, neuron zadržava sposobnost učenja i prilagođavanja.



Zatim se uzorkovanje i konvolucija ponavljaju kako bi se dodatno dodale nove značajke i slika proširila. Završni konvolucijski sloj generira sliku s 2 kanala te se konačno poziva aktivacijska funkcija tanh koja skraćuje vrijednosti od -1 do 1. Slika 5.7. prikazuje programski kod kojim je kreiran model generatora. Pomoću funkcije *generator.summary* prikazan je popis svih slojeva modela što je vidljivo na slici 5.8.

```
print('Generator Model')
init_size = X_Train.shape[1] // 4
generator = tf.keras.Sequential([
    tf.keras.layers.Dense(128 * init_size ** 2, input_dim=100, kernel_initializer=initNormal),
    BatchNormalization(),
    tf.keras.layers.Reshape((init_size, init_size, 128)),
    tf.keras.layers.UpSampling2D(size=2),
    tf.keras.layers.Conv2D(128, kernel_size=5, strides=1, padding='same', use_bias=True),
    BatchNormalization(momentum=0.8),
    LeakyReLU(0.2),
    tf.keras.layers.UpSampling2D(size=2),
    tf.keras.layers.Conv2D(64, kernel_size=5, strides=1, padding='same', use_bias=True),
    BatchNormalization(momentum=0.8),
    LeakyReLU(0.2),
    tf.keras.layers.Conv2D(2, kernel_size=5, strides=1, padding='same', use_bias=True),
    tf.keras.layers.Activation('tanh'),
])
generator.summary()
```

Slika 5.7. Model generatora

Generator Model  
 Model: "sequential\_1"

Layer (type)	Output Shape	Param #
dense_1 (Dense)	(None, 131072)	13,238,272
batch_normalization_3 (BatchNormalization)	(None, 131072)	524,288
reshape_1 (Reshape)	(None, 32, 32, 128)	0
up_sampling2d_2 (UpSampling2D)	(None, 64, 64, 128)	0
conv2d_3 (Conv2D)	(None, 64, 64, 128)	409,728
batch_normalization_4 (BatchNormalization)	(None, 64, 64, 128)	512
leaky_re_lu_2 (LeakyReLU)	(None, 64, 64, 128)	0
up_sampling2d_3 (UpSampling2D)	(None, 128, 128, 128)	0
conv2d_4 (Conv2D)	(None, 128, 128, 64)	204,864
batch_normalization_5 (BatchNormalization)	(None, 128, 128, 64)	256
leaky_re_lu_3 (LeakyReLU)	(None, 128, 128, 64)	0
conv2d_5 (Conv2D)	(None, 128, 128, 2)	3,202
activation_1 (Activation)	(None, 128, 128, 2)	0

Total params: 14,381,122 (54.86 MB)  
 Trainable params: 14,118,594 (53.86 MB)  
 Non-trainable params: 262,528 (1.00 MB)

Slika 5.8. Ispisani slojevi generatora

## 5.6. Diskriminator

Diskriminator je kreiran kao konvolucijska mreža koja koristi više slojeva kako bi izvršila ekstrakciju značajki iz ulaznih slika. Na početku se definira ulaz u prvi sloj te se model kreira funkcijom `sequential` koja omogućuje slijedno slaganje slojeva. Zatim slijedi nekoliko konvolucijskih slojeva te se u svakom povećava broj filtera (64, 128, 256, 512). Nakon svakog konvolucijskog sloja poziva se `Leaky ReLU` aktivacijska funkcija. Završni konvolucijski sloj ima samo jedan filter koji služi za procjenu je li slika stvarna ili generirana. Na posljetku, funkcijom `Flatten` izlaz se izravna u jednodimenzionalni vektor, a `sigmoid` aktivacija se koristi za binarnu klasifikaciju komprimirajući vrijednosti između 0 i 1. Slika 5.9. prikazuje programski kod kojim je kreiran model generatora. Pomoću funkcije `generator.summary` prikazan je popis svih slojeva modela što je vidljivo na slici 5.10.

```

print('Discriminator Model')
n,h,w,c=X_Train.shape
discriminator = tf.keras.Sequential()
discriminator.add(Conv2D(64, 5, padding='same', strides=(2,2),
                        input_shape=(h,w,c), kernel_initializer=initNormal, use_bias=True))
discriminator.add(LeakyReLU(0.2))
discriminator.add(Conv2D(64 * 2, 5, padding='same', strides=(2,2), use_bias=True))
discriminator.add(LeakyReLU(0.2))
discriminator.add(Conv2D(64 * 4, 5, padding='same', strides=(2,2), use_bias=True))
discriminator.add(LeakyReLU(0.2))
discriminator.add(Conv2D(64 * 8, 5, padding='same', strides=(2,2), use_bias=True))
discriminator.add(LeakyReLU(0.2))
discriminator.add(Conv2D(1, 5, padding='same', strides=(2,2), use_bias=True))
discriminator.add(Flatten())
discriminator.add(Activation('sigmoid'))
discriminator.summary()

```

Slika 5.9. Model diskriminatora

Discriminator Model  
Model: "sequential\_3"

Layer (type)	Output Shape	Param #
conv2d_11 (Conv2D)	(None, 64, 64, 64)	3,264
leaky_re_lu_8 (LeakyReLU)	(None, 64, 64, 64)	0
conv2d_12 (Conv2D)	(None, 32, 32, 128)	204,928
leaky_re_lu_9 (LeakyReLU)	(None, 32, 32, 128)	0
conv2d_13 (Conv2D)	(None, 16, 16, 256)	819,456
leaky_re_lu_10 (LeakyReLU)	(None, 16, 16, 256)	0
conv2d_14 (Conv2D)	(None, 8, 8, 512)	3,277,312
leaky_re_lu_11 (LeakyReLU)	(None, 8, 8, 512)	0
conv2d_15 (Conv2D)	(None, 4, 4, 1)	12,801
flatten_1 (Flatten)	(None, 16)	0
activation_3 (Activation)	(None, 16)	0

Total params: 4,317,761 (16.47 MB)  
Trainable params: 4,317,761 (16.47 MB)  
Non-trainable params: 0 (0.00 B)

Slika 5.10. Ispisani slojevi diskriminatora

## 5.7. Funkcija gubitka i predtreniranje diskriminatora

Nakon definiranja diskriminatora i generatora, definira se funkcija gubitka koja je u ovom slučaju binarna unakrsna entropija te se odrađuje predtreniranje diskriminatora prolazeći kroz 10 epoha. Koristeći generator, stvaraju se lažne slike na osnovu šuma *noise\_gen*. Diskriminator klasificira generirane te stvarne slike funkcijama *predictions\_fake* i *predictions\_real*. Računaju se gubitci za diskriminator na temelju njegovih predikcija. Pomoću Adam optimizatora se ažuriraju težine diskriminatora te se spremaju gubitak i točnost.

```
shuffled_indices = np.random.permutation(X_Train.shape[0])
X_Train_S = X_Train[shuffled_indices]

for i in range(X_Train_S.shape[0] // 8):
    if (i + 1) * 8 > X_Train_S.shape[0]:
        image_batch = X_Train_S[i * 8 : X_Train_S.shape[0], :, :, :]
    else:
        image_batch = X_Train_S[i * 8 : (i + 1) * 8, :, :, :]

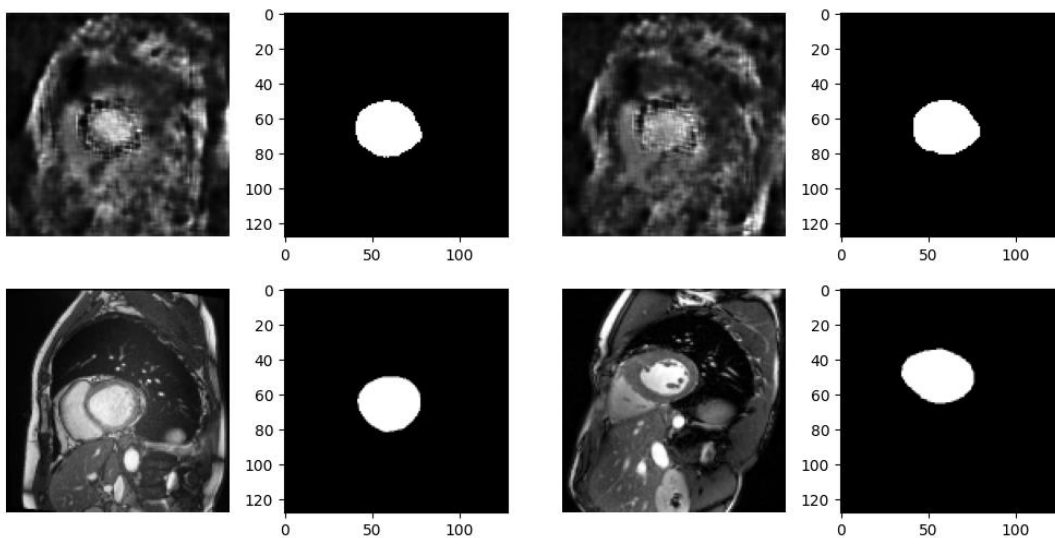
discriminator.trainable = True
generator.trainable = False
with tf.GradientTape() as d_tape:
    noise_gen = np.random.uniform(-1, 1, size=[image_batch.shape[0], 100])
    generated_images = generator(noise_gen)
    predictions_fake = discriminator(generated_images)
    y_fake = np.zeros_like(predictions_fake)
    d_loss_fake = loss_fn(y_fake, predictions_fake)
    predictions_real = discriminator(image_batch)
    y_real = np.ones_like(predictions_real)
    d_loss_real = loss_fn(y_real, predictions_real)
    d_loss = (d_loss_real + d_loss_fake) / 2
y = np.concatenate([y_fake, y_real])
predictions = np.concatenate([predictions_fake, predictions_real])
predictions = np.where(predictions < 0.5, 0, 1)
accuracy = np.mean(predictions == y)
d_grads = d_tape.gradient(d_loss, discriminator.trainable_variables)
adam_disc.apply_gradients(zip(d_grads, discriminator.trainable_variables))
losses_disc.append(d_loss.numpy())
accuracies_disc.append(accuracy)
```

Slika 5.11. Programski kod petlje kroz koju se iterira te tako predtrenira diskriminator

## 5.8. Treniranje modela i krajnji rezultat

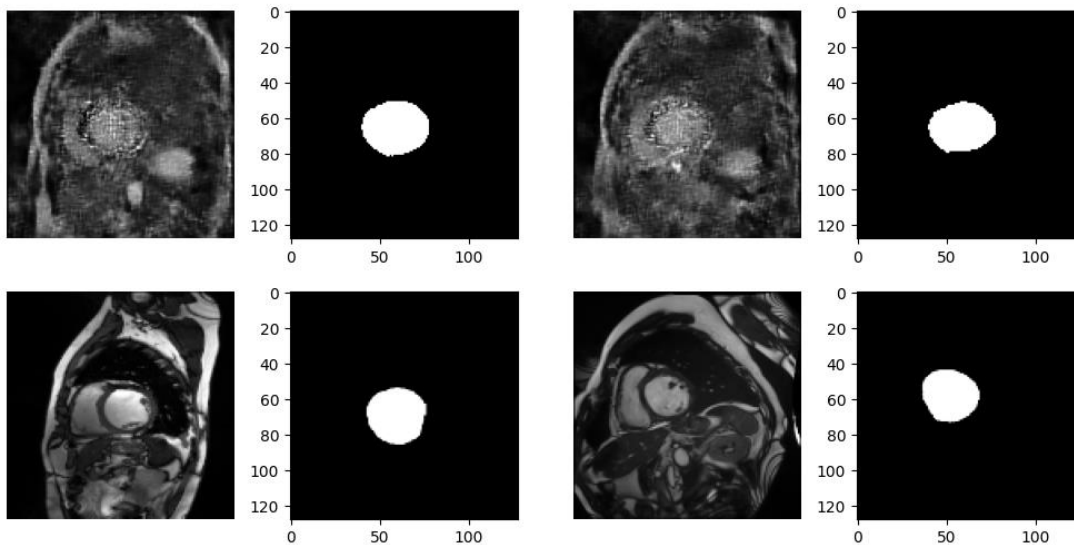
Na posljetku je pozvana funkcija *train*. Svaka njena iteracija izvršavala je sljedeće: miješanje podataka kako bi se poboljšala slučajnost, treniranje diskriminatora, ažuriranje težina generatora te prikazivanje rezultata. Broj epoha bio je 2250. U nastavku su prikazani rezultati nakon određenog broja epoha. Gornje dvije slike prikazuju generirane, umjetno kreirane snimke, a donje dvije stvarne snimke s kojima se uspoređuje kvaliteta generiranih.

- Epoha: 25
  - Prevladava šum i generirana snimka nema dovoljno kvalitetne obrise. Prikazano na slici 5.12.



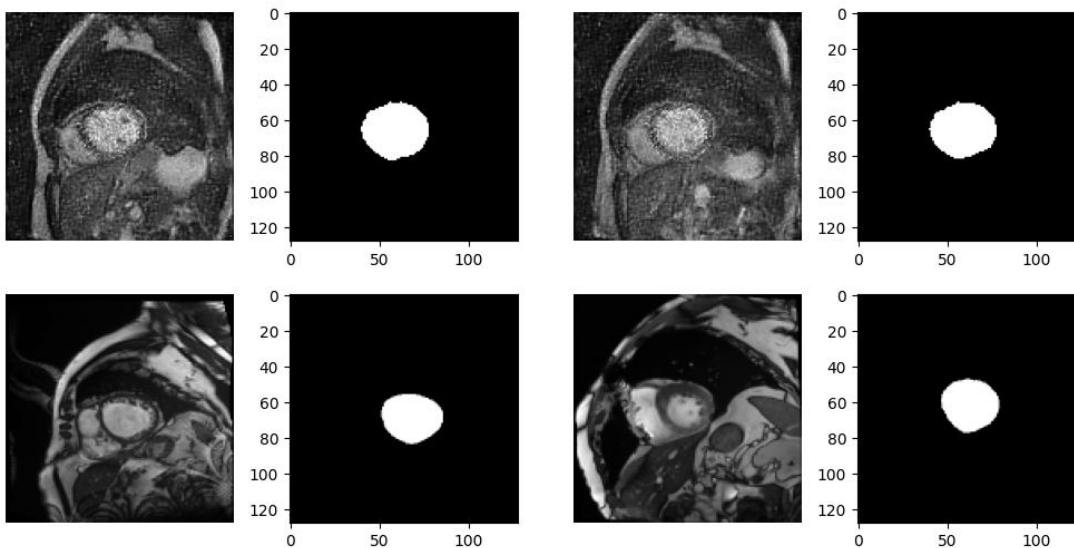
Slika 5.12. Generirane slike nakon 25 epoha (gore) u usporedbi sa stvarnim slikama (dolje)

- Epoha: 525
  - Generirana snimka dobiva oblik. Međutim, i dalje je šum izrazito prisutan. Prikazano na slici 5.13.



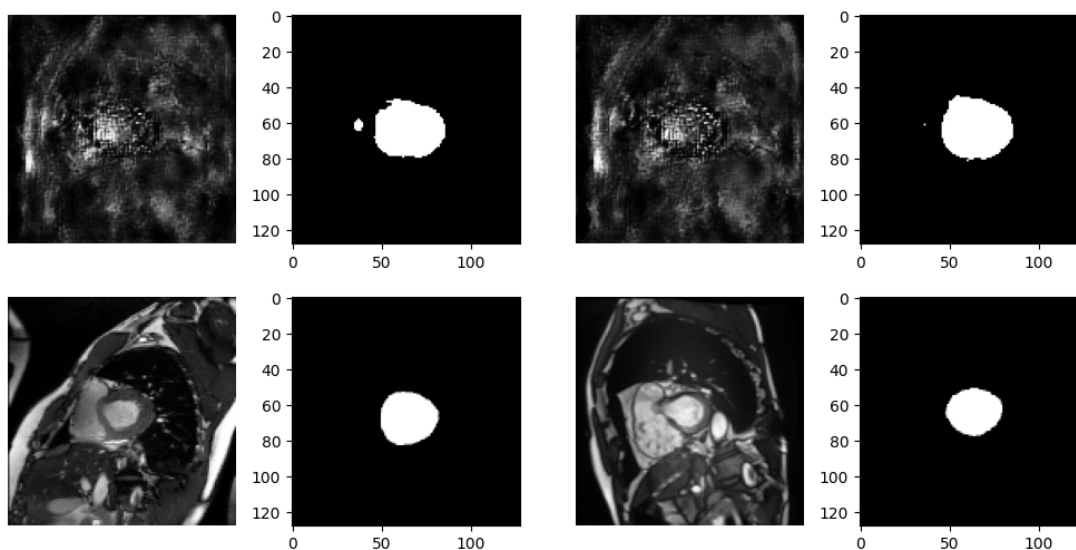
Slika 5.13. Generirane slike nakon 525 epoha (gore) u usporedbi sa stvarnim slikama (dolje)

- Epoha: 1400
  - Generirana snimka izgleda poprilično dobro, slično stvarnoj snimci. Prikazano na slici 5.14.



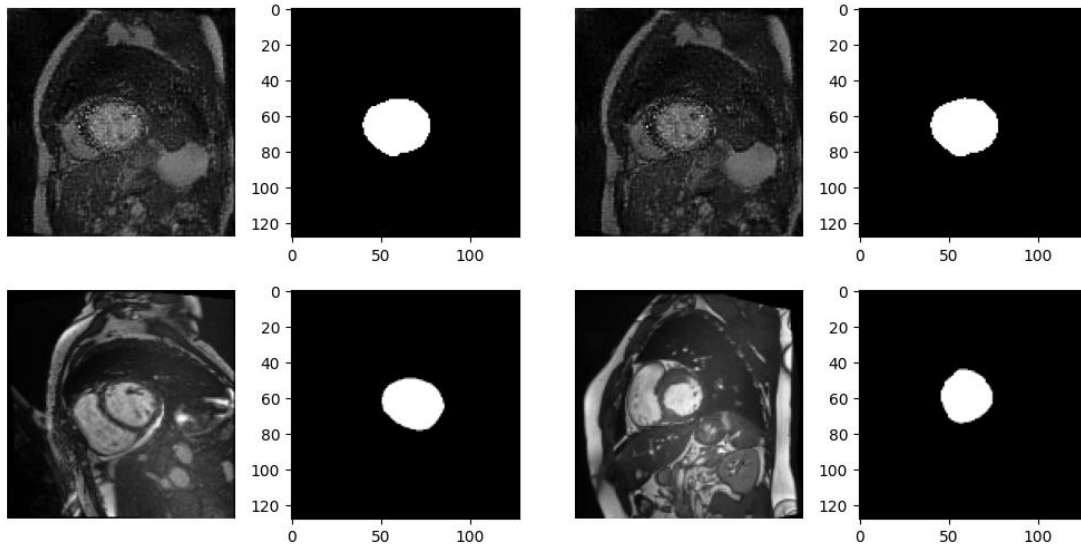
Slika 5.14. Generirane slike nakon 1400 epoha (gore) u usporedbi sa stvarnim slikama (dolje)

- Epoha: 1500
  - Zbog ulaska slika koje odudaraju od prethodnih, nove generirane slike nemaju dovoljno dobre obrise. Šum je izrazito dominantan. Prikazano na slici 5.15.



Slika 5.15. Generirane slike nakon 1500 epoha (gore) u usporedbi sa stvarnim slikama (dolje)

- Epoha: 2250
  - Generirane slike, prikazane na slici 5.16., ponovno imaju dovoljno dobar obris te su vrlo slične stvarnim slikama. S obzirom na to da se za izradu modela nije koristila naprednija verzija GAN-a, već se koristio originalni oblik koji koristi binarnu unakrsnu entropiju, te je skup podataka bio malen (345 parova snimaka i pripadnih anotiranih maski), ovakav krajnji rezultat je očekivan i zadovoljavajuć. Korištenjem naprednih, dodatno razvijenih oblika GAN-a, primjerice WGAN-a, te skupa podataka s većim brojem snimaka, generiranje bi se moglo poprilično unaprijediti.



Slika 5.16. Generirane slike nakon 2250 epoha (gore) u usporedbi sa stvarnim slikama (dolje)



## 6. ZAKLJUČAK

Generativne suparničke mreže (GAN-ovi) predstavljaju jedan od potencijalnih rješenja za problem nedovoljnog broja medicinskih snimaka i pripadajućih maski. Stvaranje umjetnog sadržaja može značajno doprinijeti medicinskoj struci jer bi korištenjem dovoljnog broja podataka mogli biti istrenirani modeli koji bi potencijalno detektirali vrlo rijetke bolesti.

Međutim, GAN-ovi nisu jednostavni za treniranje. Zahtijevaju izrazito puno vremenskih i računalnih resursa te su osjetljivi na parametre. Čak i mala promjena vrijednosti jednog od parametara može značajno narušiti kvalitetu generiranja, a za ponovnu evaluaciju potrebno je čekati satima.

Model koji je opisan u ovom radu kreiran je na temelju osnovne verzije generativnih suparničkih mreža koja koristi binarnu unakrsnu entropiju kao funkciju gubitka. Takav model ima određene nedostatke poput stvaranja sličnog sadržaja te onog koji nije grafički idealan. Nadalje, za treniranje samog modela korišten je malen skup podataka. Bez obzira na očekivane nesavršenosti, generiranje snimaka zajedno s pripadajućim maskama izvedeno je uspješno. Generiranje kvalitetnijih snimaka i različitijeg sadržaja može se dobiti primjenom modela temeljenog na naprednijim oblicima GAN-a, primjerice na WGAN-u, te korištenjem skupa podataka s većim brojem snimaka.

## LITERATURA

- [1] Weng, S., “Automating Breast Cancer Detection with Deep Learning, [Online]. Dostupno na: <https://blog.insightdatascience.com/automating-breast-cancer-detection-with-deep-learning-d8b49da17950>. Pristupljeno: [10.9.2024.]
- [2] Hollander, B, “U-Net, dilated convolutions and large convolution kernels in deep learning”, [Online]. Dostupno na: <https://medium.com/@branislav.hollander/u-net-dilated-convolutions-and-large-convolution-kernels-in-deep-learning-a849f06ffb82>. Pristupljeno: [10.9.2024.]
- [3] Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks. *CoRR*, *abs/1511.06434*.
- [4] Sauer, A., Karras, T., Laine, S., Geiger, A., & Aila, T. (2023). StyleGAN-T: Unlocking the Power of GANs for Fast Large-Scale Text-to-Image Synthesis. *International Conference on Machine Learning*.
- [5] Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-To-Image Translation with Conditional Adversarial Networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 Lipanj 2017
- [6] Piffer, S., Ubaldi, L., Tangaro, S.S., Retico, A., & Talamonti, C. (2024). Tackling the small data problem in medical image classification with artificial intelligence: a systematic review. *Progress in Biomedical Engineering*, 6.
- [7] Deng, L. (2012). The MNIST Database of Handwritten Digit Images for Machine Learning Research [Best of the Web]. *IEEE Signal Processing Magazine*, 29, 141-142.
- [8] Huang, Z. i ostali, “Fusion High-Resolution Network for Diagnosing ChestX-ray Images”, [Online]. Dostupno na: [https://www.researchgate.net/publication/338692980\\_Fusion\\_High-Resolution\\_Network\\_for\\_Diagnosing\\_ChestX-ray\\_Images](https://www.researchgate.net/publication/338692980_Fusion_High-Resolution_Network_for_Diagnosing_ChestX-ray_Images). Pristupljeno: [10.9.2024.]
- [9] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M.S., Berg, A.C., & Fei-Fei, L. (2014). ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 115, 211 - 252.
- [10] Rajpurkar, P., Irvin, J.A., Zhu, K., Yang, B., Mehta, H., Duan, T., Ding, D.Y., Bagul, A., Langlotz, C., Shpanskaya, K.S., Lungren, M.P., & Ng, A. (2017). CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning. *ArXiv*, *abs/1711.05225*.
- [11] Chaturvedi, N. “Custom Layers in Keras”, [Online]. Dostupno na: <https://medium.datadriveninvestor.com/custom-layers-in-keras-de5f793217aa>. Pristupljeno: [10.9.2024.]
- [12] Setio, A.A., Traverso, A., Bel, T.D., Berens, M.S., Bogaard, C.V., Cerello, P., Chen, H., Dou, Q., Fantacci, M.E., Geurts, B., Gugten, R.V., Heng, P., Jansen, B., Kaste, M.M., Kotov, V., Lin,

- J.Y., Manders, J., Sónora-Mengana, A., García-Naranjo, J.C., Prokop, M., Saletta, M., Schaefer-Prokop, C., Scholten, E.T., Scholten, L., Snoeren, M.M., Torres, E.L., Vandemeulebroucke, J., Walasek, N., Zuidhof, G.C., Ginneken, B.V., & Jacobs, C. (2016). Validation, comparison, and combination of algorithms for automatic detection of pulmonary nodules in computed tomography images: The LUNA16 challenge. *Medical Image Analysis*, 42, 1–13.
- [13] Karantanas, A.H., Komnos, A., Paterakis, K., & Hadjigeorgiou, G.M. (2005). Differences between CT and MR imaging in acute closed head injuries. *Cmig Extra: Cases*, 29, 1-8.
- [14] Kocak, M., “Kompjuterizirana tomografija”, [Online]. Dostupno na: <https://hemed.hr/Default.aspx?sid=19472>. Pristupljeno: [7.9.2024.]
- [15] “Computed tomography of the abdomen and pelvis”, Wikipedija, Pristupljeno: [7.9.2024.]
- [16] “CT Scans VS MRI Scans: What are the Differences Between Them”, [Online]. Dostupno na: <https://kiranpetct.com/ct-scans-vs-mri-scans-what-are-the-differences-between-them/>. Pristupljeno: [7.9.2024.]
- [17] Kolb, B., Muhammad, A., Gibb, R., “Searching for factors underlying cerebral plasticity in the normal and injured brain, *Journal of Communication Disorders*”, 2010.
- [18] Pramoditha, R., “The Concept of Artificial Neurons (Perceptrons) in Neural Networks”, Prosinac, 2021., [Online]. Dostupno na: <https://towardsdatascience.com/the-concept-of-artificial-neurons-perceptrons-in-neural-networks-fab22249cbfc>. Pristupljeno: [10.9.2024.]
- [19] Kalyan, G. R., “Deep Neural Networks”, Ožujak 2024., [Online]. Dostupno na: <https://www.linkedin.com/pulse/deep-neural-networks-ramesh-kalyan-g-xx4xc>. Pristupljeno: [10.9.2024.]
- [20] Hong, M., Rim, B., Lee, H., Jang, H., Oh, J., & Choi, S. (2021). Multi-Class Classification of Lung Diseases Using CNN Models. *Applied Sciences*.
- [21] Mall, P.K., Singh, P., Srivastav, S., Narayan, V., Paprzycki, M., Jaworska, T., & Ganzha, M. (2023). A comprehensive review of deep neural networks for medical image processing: Recent developments and future opportunities. *Healthcare Analytics*.
- [22] “Helping to Detect Breast Cancer with Artificial Intelligence”, [Online]. Dostupno na: <https://medical-professionals.com/en/detecting-breast-cancer-with-artificial-intelligence/>. Pristupljeno: [10.9.2024.]
- [23] Del Pra, Marco, “Generative Adversarial Networks”, [Online]. Dostupno na: <https://medium.com/@marcodelpra/generative-adversarial-networks-dba10e1b4424>. Pristupljeno: [14.9.2024.]

- [24] Goodfellow, Ian & Pouget-Abadie, Jean & Mirza, Mehdi & Xu, Bing & Warde-Farley, David & Ozair, Sherjil & Courville, Aaron & Bengio, Y.. (2014). Generative Adversarial Networks. *Advances in Neural Information Processing Systems*. 3. 10.1145/3422622.
- [25] Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein GAN. *ArXiv, abs/1701.07875*.
- [26] Saad, M.M., O'Reilly, R., & Rehmani, M.H. (2022). A Survey on Training Challenges in Generative Adversarial Networks for Biomedical Image Analysis. *ArXiv, abs/2201.07646*.
- [27] Xu, B., Wang, N., Chen, T., & Li, M. (2015). Empirical Evaluation of Rectified Activations in Convolutional Network. *ArXiv, abs/1505.00853*.
- [28] “Can you tell us the advantages of using leaky ReLU over ReLU?”, [Online]. Dostupno na: <https://medium.com/aaweg-i-nterview/can-you-tell-us-the-advantages-of-using-leaky-relu-over-relu-280cb27eda9e>. Pristupljeno: [14.9.2024.]
- [29] Ruby, U., Yendapalli, V., “Binary cross entropy with deep learning technique for Image classification”, 2020.

## **Sažetak**

Nedovoljno velik skup podataka jedan je od ključnih izazova kod dubokog učenja, posebno kada je riječ o medicinskim snimcima i pripadajućim im maskama. Zbog ograničenog broja snimaka, postaje vrlo izazovno razviti kvalitetne modele za detekciju rijetkih bolesti. Generiranje umjetnog sadržaja predstavlja jedno od mogućih rješenja za taj problem. U ovom radu obrađena je tematika generiranja novih medicinskih snimaka i pripadajućih im maski pomoću generativne suparničke mreže. Navedeni su i nedostatci tog modela, primjerice zahtjevnost u pogledu vremenskih i računalnih resursa, osjetljivost na parametre te sklonost kreiranju sličnog sadržaja. U završnom dijelu rada predstavljen je model osnovne verzije GAN-a primijenjen za generiranje medicinskih snimaka i maski, prikazani su rezultati generiranja te su istaknuti nedostatci tog modela.

## **Ključne riječi**

Duboke neuronske mreže, duboko učenje, generativne suparničke mreže, obrada medicinskih snimaka

## **Abstract**

### **Generative Adversarial Networks for Artificial Generation of Medical Images**

Insufficiently large datasets are one of the key challenges in deep learning, especially when it comes to medical images and their corresponding masks. Due to the limited number of images, it is very challenging to develop quality models for detecting rare diseases. Generating artificial content represents one possible solution to this problem. This paper explores the generation of new medical images and their corresponding masks using generative adversarial networks (GANs). The limitations of this model are also discussed, including the high demands in terms of time and computational resources, sensitivity to parameters, and tendency to produce similar content. In the concluding section of the paper, a model based on the basic version of GAN is presented for generating medical images and masks, along with the generated results and the highlighted shortcomings of this model.

### **Keywords**

Deep learning, deep neural networks, generative adversarial networks, medical image processing