Univerzitet u Beogradu - Elektrotehnički fakultet

Katedra za signale i sisteme

DIPLOMSKI RAD

GENERISRANJE LICA POMOĆU VARIJACIONIH AUTOENKODERA I GENERATIVNIH SUPROTSTAVLJENIH MODELA

Kandidat
Kosta Jovanović, br. indeksa 2016/0199

Mentor
dr Predrag Tadić, docent

Beograd, *septembar* 2020. godine

PREDGOVOR

Ovaj rad baziran je na znanju stečenom na predmetu Veštačka inteligencija (šifra predmeta je 13E054VI), koji je pod vođstvom dr Predraga Tadića i saradnika u nastavi Marije Milutinović, kao i na znanju stečenom istraživanjem i proučavanjem vannastavne literature. U izradi rada učestvovao je student Kosta Jovanović, uz konsultacije i savete mentora dr Predraga Tadića.

Rad se bavi interesantnim problemom generisanja slika lica. U radu će biti predstavljena dva modela, varijacioni autoenkoder i generativni suprotstavljeni model, koji uspešno generišu slike lica zadovoljavajućeg kvaliteta. Iako su modeli u ovom radu kreirani za konkretan problem, lako se mogu primeniti i na druge probleme generisanja podataka. Možda na prvi pogled izgleda kao da rešavanje ovog problema nema nekih praktičnih primena, osim povećanja skupa podataka (eng. *data augmentation*), to u realnosti nije tako. Od kolorizacije slika, povećanja rezolucije, detekcije anomalija, stilizovanja fotografija, pa sve do rešavanja suprotstavljenih primera (eng. *adversarial examples*) i simulacije okruženja kompjuterskih igara (eng. *game engine*), primene su raznorazne.

REZIME RADA

Ovaj rad se bavi problemom generisanja novih slika lica koja ne postoje u originalnom skupu podataka. Problem je rešen implementacijom dva modela, varijacionih autoenkodera i generativnih suprotstavljenih modela.

U radu su prvo predstavljene teorijske osnove navedenih modela. Prikazani su nedostaci oba modela i način na koji se mogu otkloniti. Objašnjena je prednost varijacionog autoenkodera u odnosu na klasičan, detaljno je opisan njegov način rada i algoritam treniranja. Takođe, detaljno je opisan klasičan generativni suprotstavljeni model. Objašnjene su mane klasične kriterijumske funkcije, način na koji se uvodi nova Wasserstein kriterijumska funkcija sa penalizovanjem gradijenata i izneseni su razlozi zašto ona predstavlja poboljšanje.

Zatim su modeli implementirani na skupu podataka koji sadrži slike lice i labelerane atribute (kao što su osmeh, pol, starost, itd...) koji se ne koriste za treniranje modela. Upoređen je kvalitet slika koje su modeli generisali u pogledu oštrine i sadržaja. Analizirana je raspodela dimenzija skrivenog prostora varijacionih autoenkodera, kao i zajednička raspodela. Pomoću labeliranih atributa određeni su jedinični vektori obeležja skrivenog prostora oba modela i pokazano je kako se pomoću tih vektora mogu stilizovati slike. Postepenim prelaskom iz jednog lica u drugo analizirane su osobine kontinualnosti i kompletnosti skrivenog prostora. Prikazana je stabilnost treniranja Wasserstein generativnog suprotstavljenog modela sa penalizovanjem gradijenata.

ZAHVALNICA

Posebno se zahvaljujem mentoru dr Predragu Tadiću na izdvojenom vremenu i savetima koji su doprineli izradi ovog rada. Takođe, zahvaljujem se svojoj porodici na mentalnoj i finansijskoj podršci tokom mog studiranja.

Kosta Jovanović

U Beogradu, *septembar* 2020.

SADRŽAJ

PREDGOVOR 2

REZIME RADA 3

ZAHVALNICA 4

SADRŽAJ 5

1 UVOD 7

2 METODOLOGIJA RADA 8

2.1 Varijacioni autoenkoder 8

2.2 Generativni suprotstavljeni model 12

2.2.1 Wasserstein kriterijumska funkcija 14

2.2.2 Penalizovanje gradijenata 16

2.2.3 Kompletan algoritam 17

3 REZULTATI I DISKUSIJA 20

3.1 Varijacioni autoenkoder 20

3.1.1 Rekonstrukcija originalnih slika 20

3.1.2 Generisanje novih lica 21

3.1.3 Analiza raspodele skrivenog prostora Z 21

3.1.4 Manipulacija vektorima skrivenog prostora Z 22

3.1.5 Prelazak iz jednog lica u drugo 24

3.2 Generativni suprotstavljeni model 24

3.2.1 Generisanje novih lica 24

3.2.2 Manipulacija vektorima u skrivenom prostoru Z 25

3.2.3 Prelaz iz jednog lica u drugo 26

3.2.4 Stabilnost treniranja modela 26

4 HIPERPARAMETRI 27

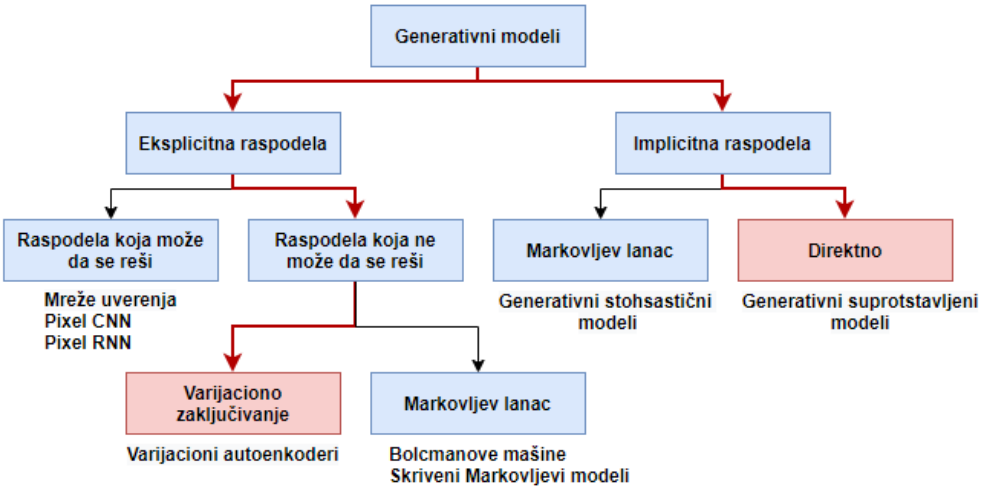
5 ZAKLJUČAK.....31

6 LITERATURA 32

PRILOG A 33

1 UVOD

Varijacioni auteonkoderi i generativni suprotstavljeni modeli spadaju u grupu generativnih modela. To su modeli koji imaju sposobnost generisanja podataka koji su najsličniji opserviranim podacima, tako što direktno ili indirektno aproksimiraju njihovu raspodelu. Oni predstavljaju pandan diskriminativnim modelima, koji umesto raspodele opserviranih podataka direktno "uče" da razlikuju podatke koji nemaju istu fundamentalnu strukturu kao opservirani podaci. Ovi modeli nisu jedini generativni modeli, kao što se može i videti sa slike 1, ali spadaju u jedne od najmoćnijih.



Slika 1: Podela generativnih modela

U radu će biti prikazana sposobnost ovih modela da generišu slike lica. Cilj je generisanje lica koja ne postoje u originalnom skupu podataka, kako bi se pokazalo da su modeli sposobni da nauče glavne karakteristike ovog skupa i da pomoću tih karakteristika sintetišu novi podatak koji će oslikavati originalni skup.

Da bi se postigla stabilnost prilikom treniranja i otklonili nedostaci klasičnih generativnih suprotstavljenih modela [1], biće uvedena Wasserstein kriterijumska funkcija (po uzoru na [2]) sa penalizovanjem gradijenata (po uzoru na [3]).

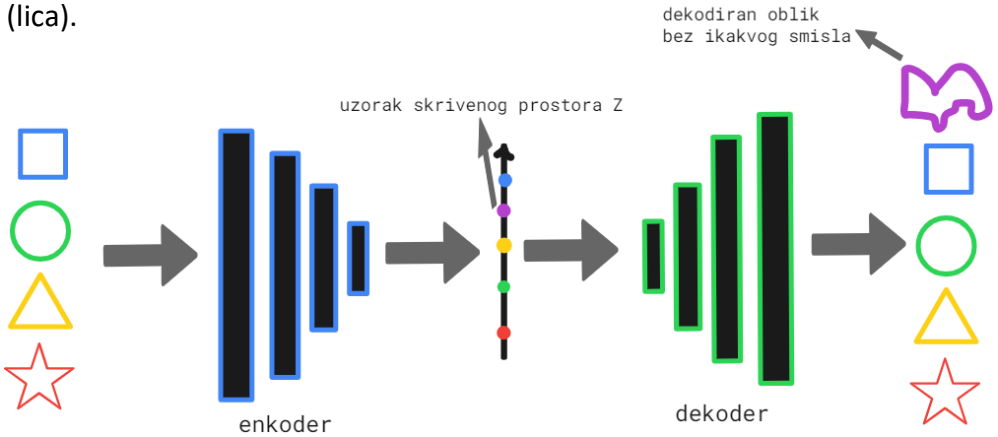
Postoji dosta tehnološki naprednijih (eng. *state of the art*) modela od ova dva koja će biti predstavljena u radu. Radovi koji se posebno ističu su: [4] "Generating Diverse High-Fidelity images with VQ-VAE-2" i [5] "Analyzing and Improving the Image Quality of StyleGAN".

U poglavlju METODOLOGIJA RADA biće objašnjen način rada i teorijske osnove modela. U poglavlju REZULTATI I DISKUSIJA biće prikazani i razmatrani rezultati dobijeni treniranjem modela na konkretnom skupu podataka koji se sastoji od slika lica. Svi hiperparametri potrebni za treniranje modela se mogu naći u poglavlju HIPERPARAMETRI.

2 METODOLOGIJA RADA

2.1 Varijacioni autoenkoder

Autoenkoder je model koji nenadgledanim učenjem efikasno kompresuje (kodira) podatke i koji na osnovu kompresovane reprezentacije može da ih rekonstruiše. Sastoji se od enkodera, funkcije kora mapira ulazne podatke (X) u "skriveni" kompresovani prostor (Z), i od dekodera, funkcije koja na osnovu kompresovanog prostora (Z) rekonstruiše ulazne podatke. Pošto se često zahteva kompleksna kompresija i rekonstrukcija ulaza, enkoder i dekoder su predstavljeni veštačkim neuralnim mrežama. Treniranje autoenkodera je jednostavno. Za kriterijumsku funkciju se uzme srednja kvadratna vrednost razlike između ulaza i rekonstrukcije, koja se minimizuje metodom "spuštanja" gradijenata unazad. Enkoder se može posmatrati kao reduktor dimenzija. Lako se dokazuje da, ako se za enkoder i dekoder uzmu jednoslojne neuralne mreže sa linearnim aktivacijama, dobija se reduktor dimenzija koji je sličan metodi glavnih komponenti (PCA). Ovakav model bi našao najbolji linearni skriveni prostor Z na koji bi projektovao podatke, ali dimenzije u tom prostoru ne bi morale da budu nezavisne. Ako se autoenkoder istrenira na skupu podataka koji sadrži lica, u skrivenom prostoru Z bi se nalazila najvažnija obeležja potrebna za rekonstrukciju lica. Intuitivno se postavlja pitanje da li se slučajnim uzimanjem uzorka skrivenog prostora Z i propuštanjem kroz dekoder može generisati novo lice? S obzirom na to da kod autoenkodera raspodela skrivenog prostora Z zavisi od dimenzija tog prostora, inicijalne raspodele podataka i arhitekture enkodera, ne postoji garancija da će skriveni prostor biti regularan (kontinualan i kompletan) (slika 2). Ovo je i logično jer u modelu autoenkodera nije implementirano ništa što bi osiguralo regularnu raspodelu skrivenog prostora Z, već je autoenkoder samo naučen da dobro kodira i dekodira sa što manjom razlikom između ulaza i rekonstrukcije. Dakle odgovor na prethodno postavljeno pitanje je ne, jednostavan autoenkoder se ne može koristiti za generisanje novih podataka (lica).



Slika 1: Ilustracija loše rekonstrukcije prilikom postojanja neregularnog skrivenog prostora

Da bi se skriveni prostor Z regularizovao, odnosno da bi se osigurala njegova pravilna raspodela (što bi omogućilo generisanje novih podataka uzorkovanjem odbiraka skrivenog prostora i propuštanjem kroz dekoder), pristupa se projektovanju varijacionog autoenkodera.

Varijacioni autoenkoder predstavlja "probabilistički spin jednostavnog autoenkodera" [6]. To je Bajesovska neuralna mreža kod koje enkoder i dekoder nisu deterministički već je enkoder definsian kao $p(z|x)$, raspodela kodirane varijable Z kada je data dekodirana varijabla X, a dekoder je definisan kao $p(x|z)$, raspodela dekodirane varijable X kada je data kodirana varijabla Z [7]. Generisanje novih podataka se vrši uzorkovanjem $z^{(i)}$ iz apriorne raspodele $p(z)$ (željene raspodele skrivenog prostora Z), a zatim uzorkovanjem $x^{(i)}$ iz uslovne raspodele $p(x|z)$. Za apriornu raspodelu $p(z)$ može se uzeti neka jednostavna raspodela. To je razumna pretpostavka jer se u skrivenom prostoru Z nalaze obeležja visokog nivoa (kao što su količina osmeha na slici, starost, itd...). Sa druge strane uslovna raspodela $p(x|z)$ je kompleksna jer se koristi za generisanje slika (u ovom slučaju slika ljudskih lica), pa nju moramo predstaviti neuralnom mrežom. Uvode se sledeće pretpostavke:

$$p(z) \equiv \mathcal{N}(0, I),$$

$$p(x|z) \equiv \mathcal{N}(m(z),\ \sigma * I),$$

gde je m(z) je deterministička funkcija (neuralna mreža) koju ćemo posle odrediti. Pošto je cilj da se na osnovu ulaznih podataka generišu novi podaci koji dolaze iz iste raspodele maksimizuje se verodostojnost ulaznih podataka $p(x) = \int p(x|z) * p(z)\, dz$. Iako za svaku vrednost z i x možemo sračunati $p(x|z)$ i $p(z)$, jer su već pretpostavljene ove raspodele, zbog visoke dimenzionalnosti prostora ovaj integral je nemoguće rešiti za svaku vrednost z. Dakle verodostojnost se ne može direktno maksimizirati. Pošto $p(x)$ figuriše i u izrazu za Bajesovu teoremu aposteriorne raspoedele $p(z|x) = \frac{p(x|z)*p(z)}{p(x)}$ nemoguće je i nju odrediti. Ovaj problem je nerešiv klasičnim Bajesovskim zaključivanjem i zahteva primenu aproksimativnih tehnika koje bi aproksimirale kompleksnu raspodelu $p(z|x)$. Jedna takva tehnika se zove varijaciono zaključivanje (zbog ove tehnike varijacioni autoenkoderi se i zovu tako).

Varijaciono zaključivanje je tehnika koja se sastoji u nalaženju raspodele, od familije parametrizovanih raspodela, koja najbolje aproksimira kompleksnu ciljnu raspodelu [7]. Definiše se parametrizovana familija raspodela i optimiziju se parametri tako da se nađe raspodela koja je najbliža ciljnoj raspodeli, prema nekom definisanom kriterijumu. Za aproksimaciju $p(z|x)$ bira se familija Gausovskih raspodela $q(z|x)$ čiji su parametri srednja vrednost i kovarijaciona matrica: $q(z|x) \equiv \mathcal{N}(n(z),\ k(z))$, gde su $n(z)$ i $k(z)$ parametrizovane funkcije (neuralne mreže). Kovarijaciona matrica raspodele $q(z|x)$ bira se da bude dijagonalna da bi se smanjio broj parametara ($2 * d$ parametara umesto $\frac{d*(d+3)}{2}$), osigurala diferencijabilnost i pozitivna definitnost kovarijacione matrice. Funkcijom $k(z)$ se

aproksimira logaritam varijanse (a ne varijansa) svake od dimenzija. Ovo je neophodno uraditi jer varijansa može biti samo pozivina, a izlaz neuralne mreže prirodno može uzeti bilo koju vrednost od $-\infty$ do $+\infty$.

Sada kada postoji kompletna enkoder-dekoder mreža može se izvesti izraz za logaritam verodostojnosti ulaznih podataka:

$\log\big(p(x)\big) = \mathrm{E}_z\big[\log\big(p(x)\big)\big]$,gde je očekivanje po $z \sim q(z|x)$(jer $\log\big(p(x)\big)$ ne zavisi od z)

$$= \mathrm{E}_z\left[\log\left(\frac{p(x|z)*p(z)}{p(z|x)}\right)\right] \text{ (sledi iz Bajesove teoreme)}$$

$$= \mathrm{E}_z\left[\log\left(\frac{p(x|z)*p(z)}{p(z|x)} * \frac{q(z|x)}{q(z|x)}\right)\right]$$

$$= \mathrm{E}_z[\log(p(x|z))] - \mathrm{E}_z\left[\log\left(\frac{q(z|x)}{p(z)}\right)\right] + \mathrm{E}_z\left[\log\left(\frac{q(z|x)}{p(z|x)}\right)\right]$$

Poslednja dva izraza predstavljaju Kullback-Leiber (KL) divergenciju. Ona predstavlja meru razlike između dve raspodele.

$$\log\big(p(x)\big) = \underbrace{\mathrm{E}_z[\log(p(x|z))] - KL(q(z|x) \parallel p(z)))}_{\mathcal{L}} + \underbrace{KL(q(z|x) \parallel p(z|x))}_{\geq 0}$$

KL divergencija između $q(z|x)$ i $p(z|x)$ je nerešiva jer nam je raspodela $p(z|x)$ nepoznata, pa se zato za kriterijsmku funkciju uzimaju samo prva dva izraza $\mathcal{L} = \mathrm{E}_z[\log(p(x|z))] - KL(q(z|x) \parallel p(z))$. Kako je KL divergencija uvek veća ili jednaka od nule, maksimizuje se donja granica logaritma verodostojnosti ulaznih podataka (ELBO – *Estimated Lower BOund*) $\mathcal{L} \leq \log\big(p(x)\big)$:

$$(m^*, n^*, k^*) = \underbrace{argmax}_{m,n,k} (\mathrm{E}_z[\log(p(x|z))] - KL(q(z|x) \parallel p(z))$$

$$(m^*, n^*, k^*) = \underbrace{argmax}_{m,n,k} (\mathrm{E}_z\left[\mathrm{c} - \frac{\|x - m(z)\|^2}{2*\sigma^2}\right] - KL(q(z|x) \parallel p(z))$$

$$(m^*, n^*, k^*) = \underbrace{argmin}_{m,n,k} (\mathrm{E}_z\left[\frac{\|x - m(z)\|^2}{2 * \sigma^2}\right] + KL(q(z|x) \parallel p(z))$$

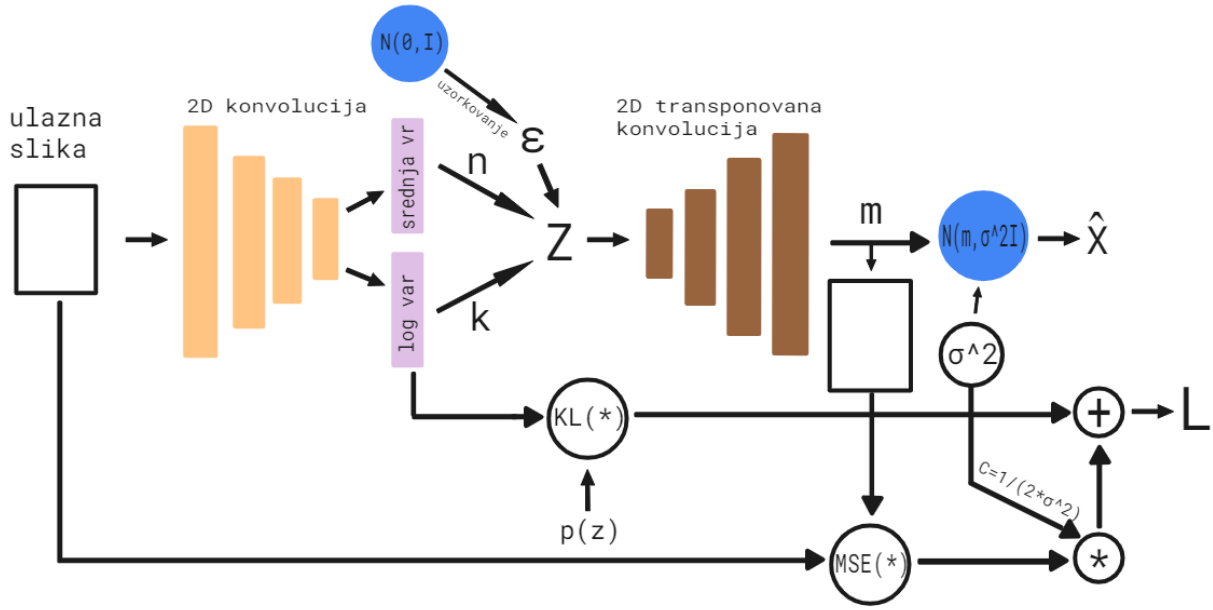
$$\boxed{(m^*, n^*, k^*) = \underbrace{argmin}_{m,n,k} (C * \mathrm{E}_z[\|x - m(z)\|^2] + KL(q(z|x) \parallel p(z)), C = const > 0}$$

U kriterijumu koji se minimizuje mogu se uočiti dva dela. Prvi deo koji predstavlja grešku prilikom rekonstrukcije i drugi, regularizacioni deo, koji predstavlja sličnost između apriorne i aposteriorne raspodele skrivenog prostora Z. Takođe može se uočiti i konstanta $C = \frac{1}{2*\sigma^2}$ pomoću koje možemo favorizovati rekonstrukciju ulaza u odnosu na regularizaciju skrivenog

prostora Z ako je C veliko, odnosno ako je varijansa oko m(z) mala. Važi i obrnuto ako je C malo.

Kao što je već napomenuto funkcije m, n i k su definisane neuralnim mrežama. Funkcije n i k nisu dve skroz odvojene neuralne mreže već dele deo arhitekture, odnosno težina. Za njihov zajednički deo projektuje se konvoluciona neuralna mreža, sa 2D konvolucionim slojevima (eng. *Conv2D layers*), koja će izdvojiti obeležja visokog nivoa sa slika lica, dok se za njihove odvojene delove uzimaju proste jednoslojne neuralne mreže. Pošto funkcija m mapira skriveni prostor Z (koji se sastoji od obeležja lica, visokog nivoa) u sliku lica, njena arhitektura se mora sastojati od slojeva koji će se postepeno povećavati. Jedan primer takvih slojeva su 2D transponovani konvolucioni slojevi (eng. *Conv2Dtranspose layers*).

Kompletan graf modela prikazan je na slici ispod.



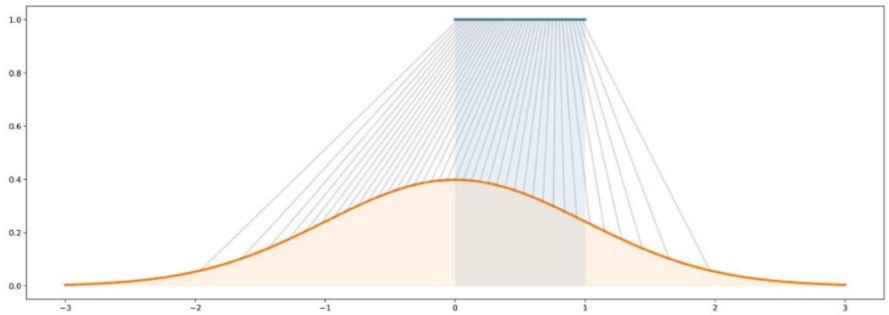
Slika 3: Kompletan graf modela varijacionog autoenkodera

Treba biti pažljiv prilikom uzorkovanja iz raspodele koju vraća enkoder ($q(z|x)$), jer se propagacija gradijenata unazad ne može izvršiti kroz stohastički čvor. U grafu iznad se ne uzorkuje direktno iz normalne raspodele $q(z|x)$, već se uzorkuje vektor iz standardne normalne raspodele pa se primenjuje reparametrizacija, u kojoj se svaka od dimenzija Z računa prema sledećoj formuli: $Z_d = \mu_d + e^{\frac{\log\_var_d}{2}} * \varepsilon_d$, gde je d dimenzija skrivenog prostora, a μ_d i $\log\_var_d$ su srednja vrednost i logaritam varijanse dimenzije d raspodele $q(z|x)$ (ovo je moguće uraditi jer je pretpostavljeno da su dimenzije skrivenog prostora Z nezavisne, odnosno jer je kovarijaciona matrica raspodele $q(z|x)$ dijagonalna).

U sekciji "Hiperparametri" mogu se naći hiperparametri neuralnih mreža, dimenzije skrivenog prostora Z, optimizator kojim minimzujemo kriterijum, itd.

2.2 Generativni suprotstavljeni model

Za razliku od varijacionih autoenkodera, koji direktno maksimizuju verodstojnost ulaznih podataka (tačnije aproksimativnu donju granicu, ELBO), generativni suprotstavljeni model ima za cilj samo generisanje novih podataka bez direktnog modelovanja kompleksnih raspodela. Pošto ne postoji direktan način za uzorkovanje iz kompleksne raspodele ulaznih podataka bez postojanja njenog eksplicitnog modela, ideja je da se uzorkuje iz neke jednostavne raspodele, kao što je na primer uniformna, i da se potom nauči funkcija koja će mapirati jednostavnu raspodelu u kompleksnu (slika 4).



Slika 4: Primer transformacije jednostavne raspodele (uniformne) u kompleksniju (Gausovu)

Ta funkcija se predstavlja neuralnom mrežom koju ćemo zvati generator. Ova mreža se trenira pomoću suprotstavljene minimax igre dva igrača iz teorije igara (odavde i potiče ime generativnih suprotstavljenih modela). Dakle u igri postoje dva igrača generator i diskriminator. Generator pokušava da generiše što realnije odbirke (slike lica) tako da prevari diskriminator, odnosno tako da ih diskriminator klasifikuje kao prave odbirke (one koji pripadaju trening skupu). Diskriminator pokušava da napravi razliku između pravih odbiraka i lažnih odbiraka koje je generisao generator. Diskriminator je takođe predstavljen neuralnom mrežom. Na osnovu navedenih suprotstavljenih ciljeva ova dva "igrača" može se definisati minimax kriterijumska funkcija:

$$\mathcal{L}(\theta_g, \theta_d) = E_x\left[\log(D_{\theta_d}(x))\right] + E_z\left[\log(1 - D_{\theta_d}\left(G_{\theta_g}(z)\right))\right],$$

gde $x \sim p_{trening}(x), z \sim p(z)$ i gde su θ_g i θ_d parametri neuralnih mreža generatora G i diskriminatora D. Izlaz diskriminatora je u granicama $[0,1]$, i predstavllja verovatnoću sa kojom je diskriminator siguran da je ulazni podatak pravi. U interesu diskriminatora je da predviđa vrednost blisku 1 za prave odbirke ($D_{\theta_d}(x) \to 1$) i vrednost blisku 0 za lažne ($D_{\theta_d}\left(G_{\theta_g}(z)\right) \to 0$). To znači da je u njegovom interesu da maksimizuje kriterijumsku funkciju. U interesu generatora je da diskriminator predviđa vrednost blisku 1 za lažne odbirke ($D_{\theta_d}\left(G_{\theta_g}(z)\right) \to 1$), dakle u njegovom interesu je da minimizira kriterijumsku funkciju:

$$(\theta_d^*, \theta_g^*) = \underbrace{max}_{\theta_d} \underbrace{min}_{\theta_g} \mathcal{L}(\theta_g, \theta_d)$$

Prilikom treniranja generativnog suprotstavljenog modela naizmenično se primenjuju:

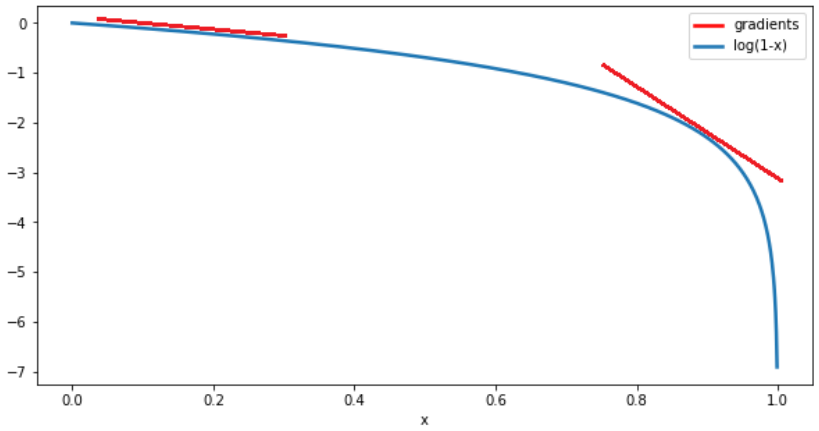
1. metoda uspona gradijenta na parametrima diskriminatora:

$$\theta_d^* = \underbrace{max}_{\theta_d} (E_x[\log(D_{\theta_d}(x))] + E_z\left[\log(1 - D_{\theta_d}\left(G_{\theta_g}(z)\right))\right])$$

2. metoda spuštanja gradijenta na parametrima generatiora:

$$\theta_g^* = \underbrace{min}_{\theta_g} (E_z\left[\log(1 - D_{\theta_d}\left(G_{\theta_g}(z)\right))\right])$$

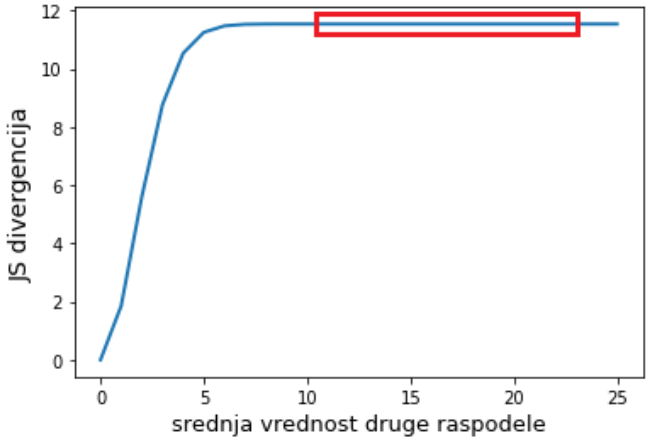
Kada se prikaže grafik funkcije $\log(1-x)$ za $x \in [0,1]$ (slika 5), može se zaključiti da kada je $x = D_{\theta_d}\left(G_{\theta_g}(z)\right)$ blisko 0, odnosno kada je odbirak klasifikovan kao lažni od strane diskriminatora, tada je gradijent mali. To čini treniranje generatora nestabilnim, jer kada generator nije uspeo da prevari diskriminator, poželjno je da on nauči nešto iz toga i da se unapredi. Takođe, kada je x blisko 1, tada je gradijent veliki, što opet nije dobro jer je u tom slučaju generator uspeo da prevari diskriminator i ne treba mnogo da se menja.



Slika 5: Grafik funkcije log(1-x) (plava boja) i gradijenti (crvena boja)

Iz ovoga sledi da kada je diskriminator optimalan (kada nema problema da razlikuje prave i lažne odbirke) i kada su $p_{trening}(x)$ i $p_{lažni}(x)$ daleko jedna od druge, tada generator skoro ništa ne uči jer dolazi do iščezavanja gradijenata. Ovo osobina je posledica činjenice da se kriterijumska funkcija, pri optimalnom diskriminatoru, može izraziti preko Jensen-Shannon (JS) divergencije (izvođenje se nalazi u prilogu A): $\mathcal{L}(\theta_g, {\theta_d}^*) = 2 * JS(p_{trening}(x) \parallel p_{lažni}(x)) - 2 * log2$. JS divergencija predstavlja meru razlike između između dve raspodele, koja je za razliku od prethodno spomenute KL dvergencije simetrična ($JS(p,q) = JS(q,p)$). Ako se postepeno udaljavaju dve jednodimenzionalne Gausovske raspodele sa varijansama 1 (odnosno ako se prvoj raspodeli fiksira srednja vrednost na 0, a drugoj varira u granicama

$[0,25]$) i u svakoj tački odredi JS divergenciju za te dve raspodele, može se uočiti da će gradijent polako iščeznuti na velikim udaljenstima (slika 6).



Slika 6: JS divergencija za dve Gausovske raspodele sa vrijansama 1 koje se postepeno udalljavaju (prva raspodela je fiksna, dok se drugoj menja srednja vrednost u intervalu [0,25])

Ako diskriminator radi kako treba, tada gradijenti kriterijumske funkcije padnu blizu 0 i generator ima problema da bilo šta nauči. U drugom slučaju kada diskriminator nije optimalan generator nema dobru povratnu informaciju od njega, i opet ima problema da bilo šta nauči. Iz prethodne analize može se zaključiti da kriterijumska funkcija nije odgovarajuća. Da bi se otklonili prethono navedeni nedostaci i stabilizovalo treniranje generativnih suprotstavljenih modela uvodi se nova Wasserstein kriterijumska funkcija.

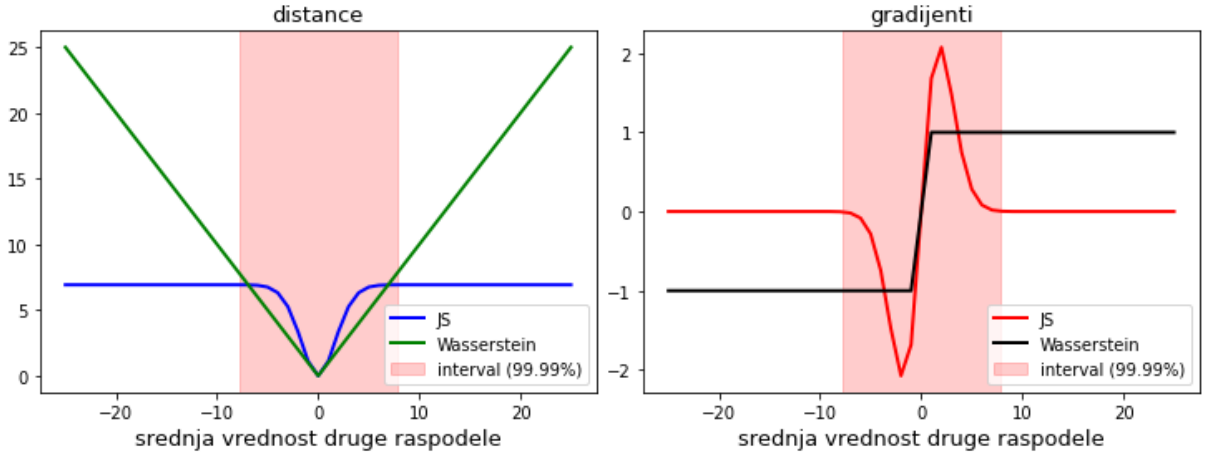
2.2.1 Wasserstein kriterijumska funkcija

Wasserstein-1 (EM – Earth Mover) distanca je metrika kojom se određuje distanca između dve raspodele na nekom zadatom metričkom prostoru. To je minimalna cena transporta mase potrebna da se jedna raspodela izjednači sa drugom, odnosno matematički:

$$W(p_{trening}, p_{lažni}) = \underbrace{inf}_{\gamma \in \Pi(p_{trening}, p_{lažni})} (E_{x,y\sim\gamma}[\|x - y\|]), \quad (1)$$

gde je Π skup svih zajedničkih raspodela $\gamma(x, y)$ čije su marginalne raspodele $p_{trening}$ i $p_{lažni}$ (intuitivno $\gamma(x, y)$ ukazuje na količinu potrebne "mase" transporta) [7]. Ova distanca ima mnogo bolje osobine, u pogledu kontinualnosti i upotrebljivosti gradijenata, od JS divergencije. Ova tvrdnja se može ilustrovati na već pomenutom primeru udaljavanja dve Gausovske raspodele sa varijansama 1. Kao i malopre, varira se srednja vrednost druge raspodele (u ovom slučaju u granicama $[-25,25]$) i za svaki pomeraj se određuje JS divergencija i EM distanca, koje su prikazane na levom grafiku slike. Na desnom grafiku mogu se videti gradijenti ovih funkcija. Jasno se vidi da su gradijenti EM distance upotrebljiviji jer

skoro nikad nisu 0 (ne dolazi do iščezavanja gradijenata) i nisu ni preveliki (ne dolazi do "eksplozije gradijenata"), što nije slučaj sa gradijentima JS divergencije. Roze bojom je slici predstavljen dvostrani interval poverenja koji obuhvata 99.99% Gausovske raspodele. Dakle van ovog intervala raspodele se ne seku. U generativnom suprotstavljenom modelu područje van intervala bi označavalo područje u kome generator loše obavlja svoj posao (jer se raspdodela generisanih podataka ne poklapa sa raspodelom pravih podataka ni u jednoj tački) i u kome bi trebalo da se unapredi. Kako je kod JS divergencije gradijent van intervala jednak 0, generator neće nimalo napredovati. To nije slučaj kod EM distance, jer je kod nje gradijent jednak ± 1. Zaključuje se da EM distanca pokriva sve navedene nedostatke JS divergencije i da je odličan kandidat za kriterijumsku funkciju. Prethodna analiza se intuitivno može primeniti i na višedimenzione raspodele.



Slika 7: Na levom grafiku su prikazane JS divergencija i EM distanca za dve Gausovske raspodele sa varijansama 1 koje se postepeno udaljavaju (prva raspodela je fiksna, dok se drugoj menja srednja vrednost u intervalu [-25,25]). Na desnom grafiku su prikazani njihovi gradijenti. Roze bojom je na graficima označen dvostruki dupli interval poverenja od 99.99%.

Izraz (1) za Wasserstein-1 distancu se ne može kao takav uzeti za kriterijumsku funkciju jer je Infinum nerešiv (nemoguće je naći sve zajedničke raspodele). Zato se izraz može pojednostovaiti pomoću Kantorovich-Rubenstein dualnosti [8]:

$$W(p_{trening}, p_{lažni}) = \frac{1}{K} \underbrace{sup}_{\|f_L\| \leq K} (E_{x \sim p_{trening}}[f(x)] - (E_{x \sim p_{lažni}}[f(x)]),$$

gde se supremum vrši nad svim K-Lipschitz funkcijama $f: X \to \mathbb{R}$. Funkcija f je K-Lipschitz ako postoji $K \geq 0$, $K \in \mathbb{R}$ takvo da je za svako $X_1, X_2 \in X$, $|f(X_1) - f(X_2)| \leq K * |X_1 - X_2|$ [2]. Odnosno funkcija f je K-Lipschitz ako joj je izvod ograničen konstantom K (ovo ne znači da je funkcija diferencijabilna u svakoj tački, jer je na primer funkcija $f(x) = |x|$ 1-Lipschitz funkcija, a nije diferencijabilna u tački $x = 0$). Uvodi se pretpostavka da funkcija f dolazi iz familije parametrizvanih funkcija f_θ. Sada se funkcija f može predstaviti neuralnom mrežom sa parametrima θ. U Wasserstein generativnom suprotstavljenom modelu funkciju f predstavlja neuralna mreža diskriminatora. Ona u ovom slučaju neće imati direktnu diskriminativnu ulogu

jer funkcija f slika X u $\mathbb{R}$, a ne X u $[0,1]$. Dakle diskriminator će sada imati ulogu da kritikuje koliko su realni ulazni podaci. Praktično gledano, ovo znači da se u poslednjem sloju neuralne mreže diskriminatora uklanja sigmoid aktivaciona funkcija. Konačno, može se napisati izraz za Wassestein kriterijumsku funkciju:

$$\boxed{\mathcal{L}(\theta_g, \theta) = E_{x\sim p_{trening}}[f_\theta(x)] - E_{x\sim p_{lažni}}\left[f_\theta\left(G_{\theta_g}(x)\right)\right]}$$

U Wassesrstein generativnom suprotstavljenom modelu diskriminator (kritika) se može trenirati do optimalnost, ovo je prednost u odnosu na model sa klasičnom kriterijumskom funkcijom kod koga dolazi do iščezavanja gradijenata. Ovo je mnogo bitna stvar jer sprečava efekat "kolapsa moda". To je efekat do koga dolazi kada generator uspe da nađe mali skup primera (modova) pomoću kojih uvek prevari diskriminator i zbog toga on nije sposoban da proizvede neki novi primer. Pošto možemo trenirati diskriminator do perfektnosti, generator se neće zaglaviti na ovom malom skupu primera.

Da bi se osiguralo da funkcija f_θ tokom treniranja bude K-Lipschitz, neophodno je odseći vrednosti težina θ u svakoj iteraciji. "Odsecanje težina je užasan način za sprovođenje K-Lipschitz. Ako je paramatar odsecanja prevelik, onda je potrebno dosta vremena težinama da dostignu svoj limit, time čineći mnogo teškim treniranje kritike do optimalnosti. Ako je odsecanje malo, ovo može lako dovesti do iščezavanja gradijenata gde je broj slojeva veliki, ili gde se ne koristi sloj za normalizaciju serije." [2]. Zbog navednih razloga, umesto odsecanja težina uvodi se penalizovanje gradijenata.

2.2.2 Penalizovanje gradijenata

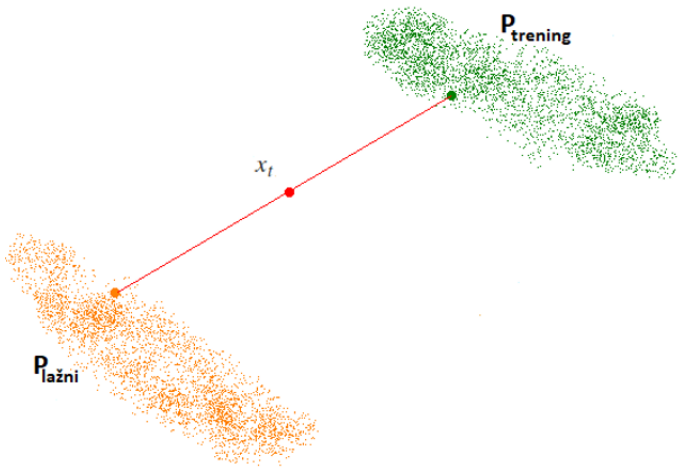
Umesto odsecanja težina, sa ciljem ograničavanja gradijenata, ova metoda kriterijumskoj funkciji dodaje regularizacioni deo koji penalizuje odstupanje norme gradijenata od neke vrednosti.

Kao što je već rečeno funkcija $f_\theta(x)$ je 1-Lipschitz ako norma gradijenata u svakoj tački nije veća od 1. Dakle norma gradijenta izlaza diskriminatora (kritike) u odnosu na ulaz se mora ograničiti. Kako je nemoguće ograničiti normu gradijenata u svakoj tački, u naučnom radu [3] je predloženo da se izvrši blaža verzija ograničenja samo nad odbricima x_t, koji predstavljaju odbirke unifrmno uzorkovane sa prave linije koja spaja prave i lažne (generisane) odbirke (slika 8). Odbirak x_t je interpolisana vrednost između pravog i lažnog odbirka, odnosno između prave i generisane slike lica. Ovo je opravdano postojanjem jediničnog gradijenta na pravama koje spajaju prave i lažne odbirke, pod uslovom da je $f_\theta(x)$ 1-Lipschitz funkcija, što sledi iz teoreme 1.

Teorema1: Neka je f_{θ^*} 1-Lipscchitz funkcija sa parametrima θ^*, koji su optimalno rešenje Wasserstein kriterijumske funkcije. Ako je f_{θ^*} diferencijabilna i ako je zajednička raspodela $\gamma(x, y)$ (čije su marginalne raspodele $p_{trening}$ i $p_{lažni}$) za $x = y$ jednaka 0, tada je:

$$(\forall x_t = t * x + (1-t) * y, 0 \leq t \leq 1)\ \ P_{x,y\sim\gamma}\left[\nabla f_{\theta^*}(x_t) = \frac{y - x_t}{\|y - x_t\|}\right] = 1.$$

Dokaz se može naći u [3].



Slika 8: Ilustracija odbirka nastalog interpolacijom pravog i lažnog odbirka. Slika je preuzeta sa [9].

Konačno, uvrštavanjem ograničenja dobijamo Wasserstein kriterijumsku funkciju sa penalizovanim gradijentima:

$$\mathcal{L}(\theta_g, \theta) = E_{x\sim p_{trening}}[f_\theta(x)] - E_{x\sim p_{lažni}}\left[f_\theta\left(G_{\theta_g}(x)\right)\right] + \lambda * E_{x_t\sim p_{x_t}}\left[\left(\left\|\nabla_{x_t} f_\theta(x_t)\right\|_2 - 1\right)^2\right]$$

gde je λ hiperparametar.

Napomena: Ako se koristi penalizovanje gradijenata, u neuralnoj mreži diskriminatora (kritike) se ne sme koristiti sloj za normalizaciju serije podataka. Taj sloj bi uveo korelaciju između podataka u jednoj seriji i kriterijum više ne bi bio validan, jer on za svaki podatak pojedinačno penalizuje normu gradijenata izlaza u odnosu na ulaz kritike.

2.2.3 Kompletan algoritam

Kao što je već rečeno, u interesu diskriminatora je da maksimizuje kriterijumsku funkciju, a u interesu generatora da je minimizuje. Da bi mogla da se primeni metoda spuštanja

gradijenata, prilikom treniranja diskriminatora prvi deo kriterijuma se mora uzeti sa predznakom minus:

1. metoda spuštanja gradijenta na parametrima diskriminatora (kritike):

$$\theta^* = \underbrace{min}_{\theta}\Big(-\Big(E_{x\sim p_{trening}}[f_\theta(x)] - E_{x\sim p_{lažni}}\Big[f_\theta\Big(G_{\theta_g}(x)\Big)\Big]\Big) + \lambda * E_{x_t\sim p_{x_t}}\Big[\Big(\big\|\nabla_{x_t} f_\theta(x_t)\big\|_2 - 1\Big)^2\Big]\Big)$$

2. metoda spuštanja gradijenta na parametrima generatiora:

$$\theta_g^* = \underbrace{min}_{\theta_g}\Big(-E_{x\sim p_{lažni}}\Big[f_\theta\Big(G_{\theta_g}(x)\Big)\Big]\Big)$$

Sada se može definisati kompletan algoritam za treniranje generatora i diskriminatora (kritike):

Algoritam: Wasserstein generativni suprotstavljeni model sa penalizovanjem gradijenata

Definiši: koeficijent za penalizovanje gradijenata λ, broj iteracija kritike u odnosu na 1 iteraciju generatora $n_{kritika}$, veličinu serije m, hiperparametre Adam optimizatora α, β_1, β_2, arhitektrue neuralnih mreža G_{θ_g} i f_θ

Inicijalizuj: parametre kritike θ, parameter generatora θ_g

while θ_g nije konvergiralo **do**
  for i=1...$n_{kritika}$ **do**
    for j =1...m **do**
      uzorkuj $x\sim p_{trening}$, $z\sim p(z)$, $t\sim U[0,1]$
      $y \leftarrow G_{\theta_g}(z)$
      $x_t \leftarrow t * x + (1-t) * y$
      $\mathcal{L}^{(j)}(\theta_g, \theta) \leftarrow -\big(f_\theta(x) - f_\theta(y)\big) + \lambda * \Big(\big\|\nabla_{x_t} f_\theta(x_t)\big\|_2 - 1\Big)^2$
    end for
    $\theta \leftarrow Adam\left(\nabla_\theta \frac{1}{m}\sum_{i=1}^{m}\mathcal{L}^{(j)}(\theta_g, \theta)\,; \alpha, \beta_1, \beta_2\right)$
  end for

11: uzorkuj seriju $\{z^j\}_{j=1}^{m} \sim p(z)$

12: $\theta_g \leftarrow Adam\left(\nabla_\theta \frac{1}{m}\sum_{i=1}^{m} -f_\theta\left(G_{\theta_g}(z)\right); \alpha, \beta_1, \beta_2\right)$

13: **end while**

Arhitekture neuralnih mreža f_θ i G_{θ_g}, kao i svi ostali hiperparametri, mogu se naći u sekciji HIPERPARAMETRI. Kada se završi treniranje neuralnih mreža, generisanje novih slika se može izvršiti uzorkovanjem vektora skrivenog prostora $z \sim p(z)$ i njegovim propuštanjem kroz generator.

3 REZULTATI I DISKUSIJA

U ovom poglavlju će biti prikazani rezultati za prethodno opisana dva modela: varijacioni autoenkoder i Wasserstein generativni suprotstavljeni model sa penalizovanjem gradijenata. Biće prikazana njihova sposobnost da generišu nove slike lice, što i predstavlja glavni cilj ovog rada. Takođe, biće analizirani skriveni prostori oba modela, kao i stabilnost i konvergencija algoritama treniranja modela.

Oba modela su trenirana na javno dostupnom skupu podataka koji se može naći u [10]. Ovaj skup podataka sastoji se od 202599 RGB slika poznatih ličnosti, koje su labelirane različitim atributima (na primer, osmeh, naočare, starost, pol,...). Ovi atributi se ne koriste prilikom treniranja, ali će biti od velikog značaja prilikom analiziranja skrivenog prostora modela.

Skipte za treniranje modela su napisane u programskom jeziku *Python*. One su pokretane na *Google Colaboratory* platformi, koja pruža 12GB ram memorije i GPU nepoznatog proizvođača (najčešće Nvidia K80s, T4s, P4s i P100s).

Svi potrebni hiperparametri se mogu naći u sekciji “Hiperparametri”.

3.1 Varijacioni autoenkoder

Slede rezultati dobijeni za model varijacionog autoenkodera koji je treniran za rezoluciju slika 128x128.

3.1.1 Rekonstrukcija originalnih slika

Slika 9: Slike ispod predstavljaju rekonstruisane verzije originalnih slika iznad, i nastale su propuštanjem originalnih slika kroz kompletan model varijacionog autoenkodera

Rekonstrukcija slike se dobija propuštanjem originalne slike kroz ceo model. Rekontruisane slike su dosta mutnije od originalnih. To je očekivani rezultat za varijacione autoenkodere jer se u kriterijumskoj funkciji usrednjava kvadratna razlika između originalnih i rekonstruisanih piksela. Jasno se može videti da je varijacioni autoenkoder naučio atribute i obeležja visokog nivoa koja karakterišu lice.

3.1.2 Generisanje novih lica

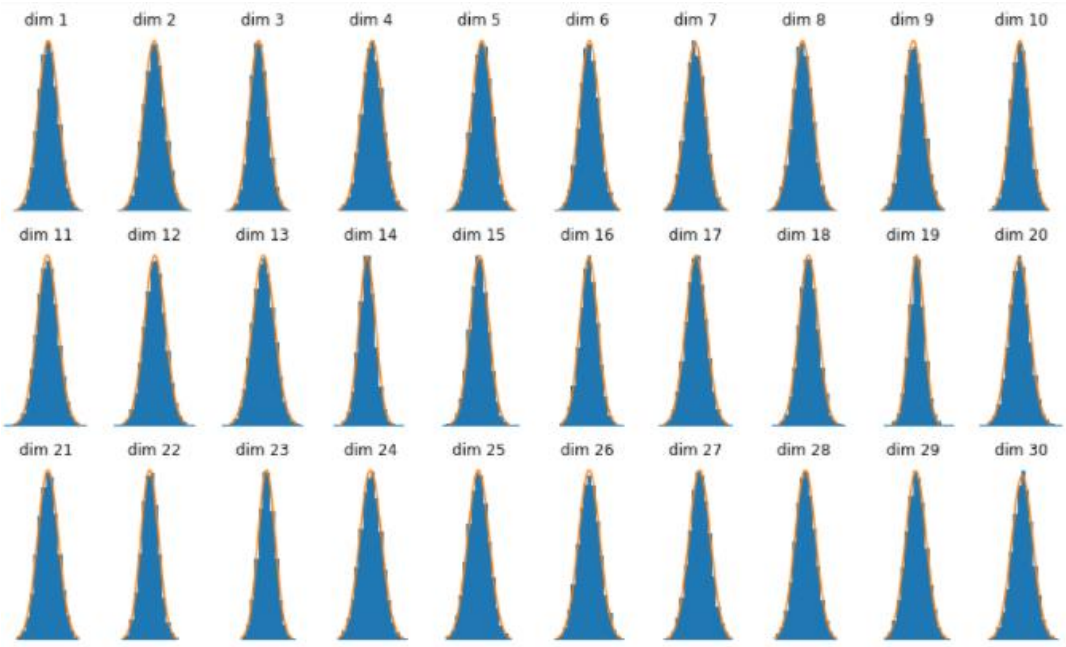
Slika 10: Generisane slike nastale uzorokovanjem prostora z iz apriorne raspodele p(z) i njegovim propuštanjem kroz dekoder

Nova lica se generišu uzorokovanjem skrivenog prostora z iz apriorne raspodele p(z) i njegovim propuštanjem kroz dekoder. Kao što se može videti model je uspeo da na osnovu uzorkovanog vektora z rekonstruiše lica, koja najverovatnije ni ne postoje u skupu podataka za treniranje.

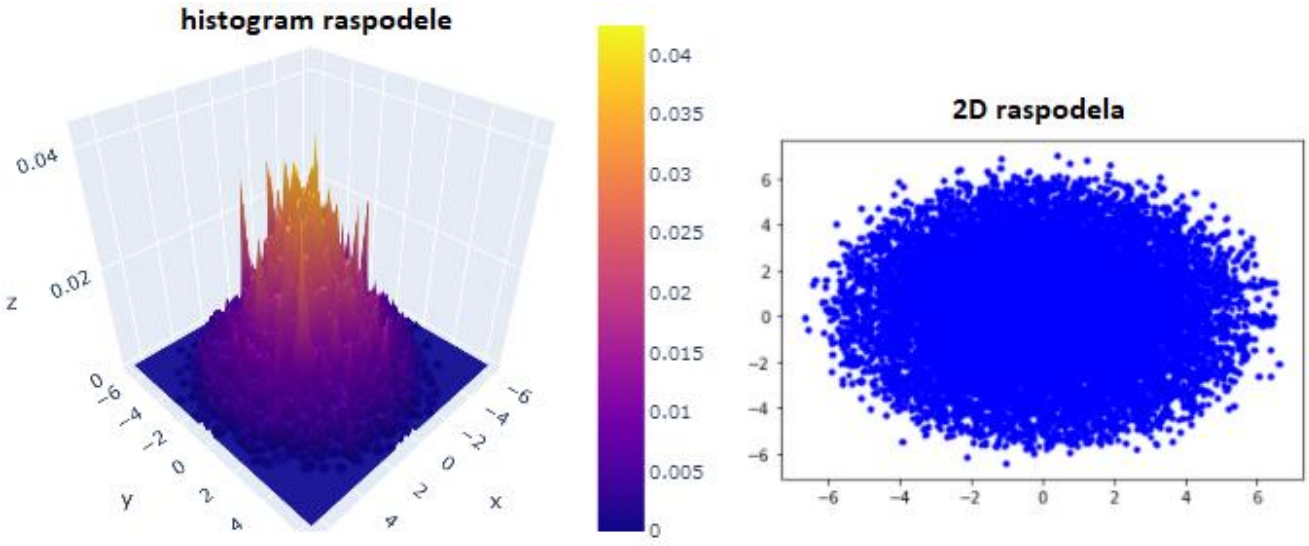
3.1.3 Analiza raspodele skrivenog prostora Z

Pošto je pretpostavljeno da je kovarijaciona matrica aposteriorne raspodele $q(z|x)$ skrivenog prostora Z dijagonalna, očekuje se da raspodela svake dimenzije ima oblik standardne normalne raspode, ako je model istreniran na pravi način. Ovo je dokazano na slici 11, na kojoj su prikazane raspodele prvih 30 dimenzija skrivenog prostora (plava boja), evaluirane na 20000 slika, i standardne normalne raspodele (crvena boja).

Pošto je nemoguće vizualizovati 200-dimenzionu raspodelu primenjena je tehnika redukcije dimenzija t-SNE [11], pomoću koje je raspodela skrivenog prostora Z redukovana na dvodimenzionu. Raspodela i histogram su prikazani na slici 12. Može se uočiti da raspodela skrivenog prostora Z stvarno ima oblik normalne raspodele.



Slika 11: Poređenje raspodela prvih 30 dimenzija skrivenog prostora Z eveluiranih na 20000 slika (plava boja) u odnosu na standardne normalne raspodele (crvena boja)



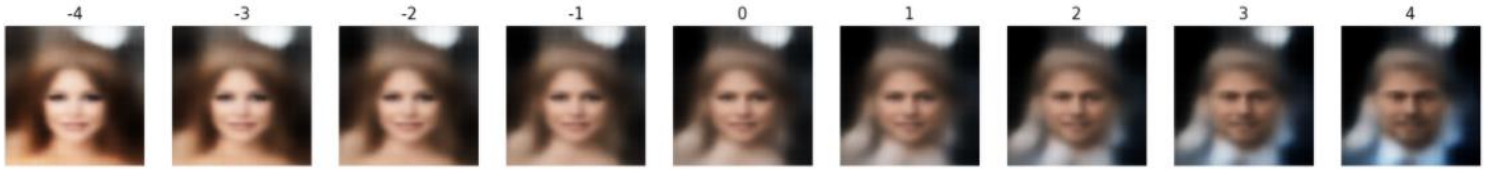
Slika 12: 2D raspodela skrivenog prostora Z posle redukcije dimenzija primenom t-SNE algoritma (grafik desno) i 3D histogram te raspodele (slika levo)

3.1.4 Manipulacija vektorima skrivenog prostora Z

Nad vektorima skrivenog prostora Z može se primeniti neka aritmetika koja će rezultovati u neki novi vektor, koji će kada se dekoduje u sliku imati vizuelnu reprezantaciju. Na primer, moguće je izdvojiti vektor koji je zadužen za postojanje naočara i dodati ga na kodirani vektor slike lica koje nema naočare. Dobijeni vektor provučen kroz dekoder bi proizveo istu sliku lica

koje bi sada imalo i naočare. Vektor koji je zadužen za postojanje naočara može se odrediti tako što se uzme srednja vrednost vektora svih slika koje sadrže naočare i oduzme od srednje vrednost vektora svih slika koje ne sadrže naočare (ovde su od velike koristi labelirani atributi). Normalizacijom dobijenog vektora bi se odredio jedinični vektor koji pokazuje u pravcu: ne postoje naočare → postoje naočare. Matematički gledano u skrivenom prostoru se primenjuje sledeća aritmetika: $z_{novo} = z_{staro} + \alpha * \vec{o}$, gde je α konstanta koja određuje intenzitet jediničnog vektora obeležja $\vec{o}$ koji se dodaje.

$$\vec{o} = \check{z}ena \rightarrow muškarac$$



Slika 13: Izdvajanje jediničnog vektora zaduženog za pol

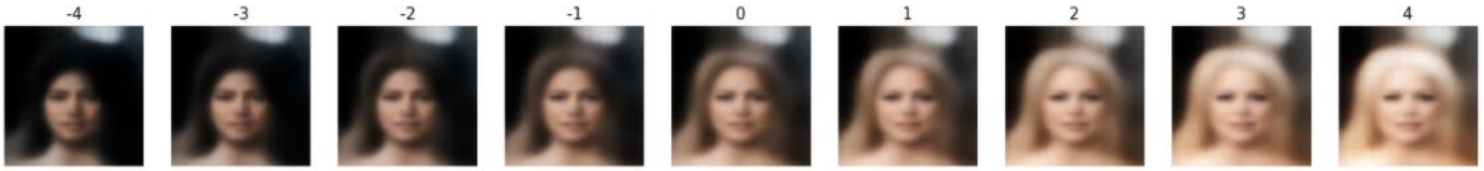
$$\vec{o} = bez\ osmeha \rightarrow sa\ osmehom$$

Slika 14: Slika 13: Izdvajanje jediničnog vektora zaduženog za osmeh

$$\vec{o} = bez\ naočara \rightarrow sa\ naočarima$$

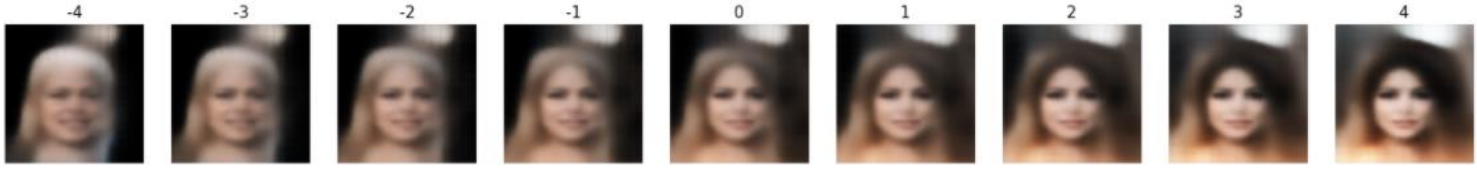
Slika 15: Izdvajanje jediničnog vektora zaduženog za naočare

$$\vec{o} = crna\ kosa \rightarrow plava\ kosa$$



Slika 16:Izdvajanje jediničnog vektora zaduženog za boju kosu

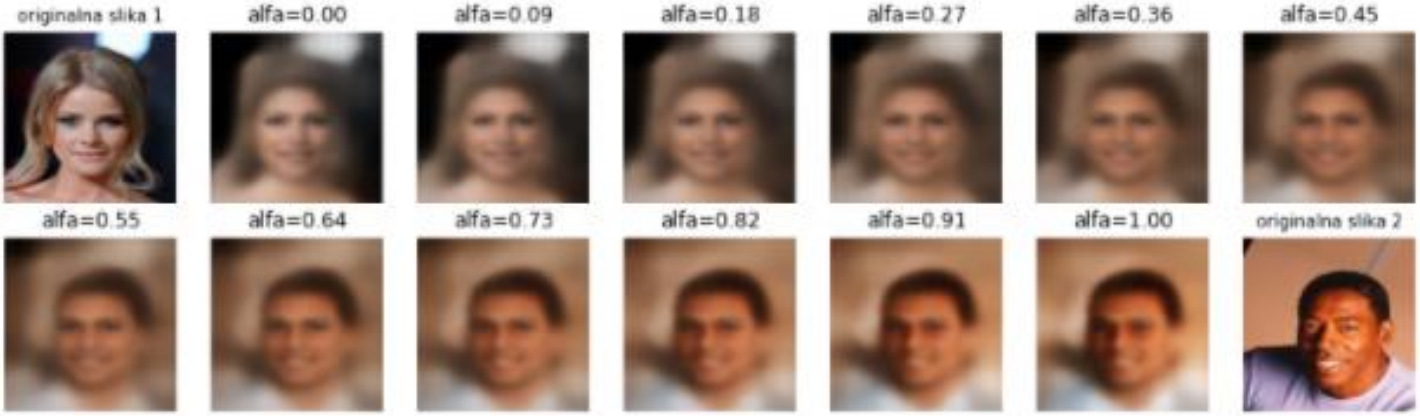
$$\vec{o} = staro \rightarrow mlado$$



Slika 17: Izdvajanje jediničnog vektora zaduženog za starost

3.1.5 Prelazak iz jednog lica u drugo

Neka postoje dva vektora, A i B, u skrivenom prostoru Z, koja dekodiranjem rezultuju u dve slike lica. Ako je skriveni prostor kontinualan, moguće je "prošetati" od tačke A do tačke B i dekodirati svaku tačku između. Rezultat dekodiranja bi bio postepeni prelaz sa jedne slike na drugu. "Šetanje" između vektora se vrši po formuli: $z = z_A * (1 - \alpha) + z_B * \alpha$



Slika 18: Postepeni prelazak iz jednog lica u drugo u 12 tačaka

3.2 Generativni suprotstavljeni model

Slede rezultati za Wasserstein generativni suprotstavljeni model sa penalizovanjem gradijanata za rezoluciju slika 64x64.

3.2.1 Generisanje novih lica

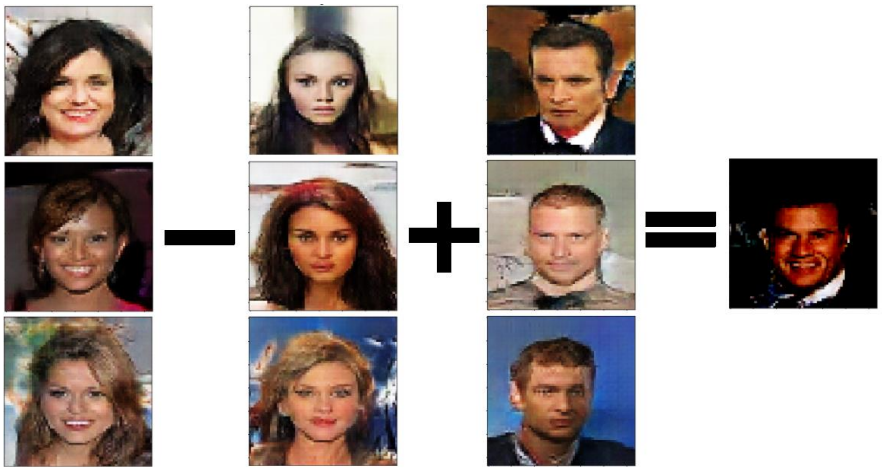
Nova lica se mogu generisati uzorkovanjem vektora skrivenog prostora Z iz apriorne raspodele p(z) i njegovim propuštanjem kroz generator. Kao što se može videti ovaj model je uspeo da pravilno generiše lica (koja najverovatnije ni ne postoje u skupu podataka). Slike nisu mutne i model je uspeo da nauči obeležja lica visokog nivoa.

Slika 19: Generisanje novih slika lica uzorokovanjem vektora skrivenog prostora Z iz apriorne raspodele p(z) i njegovim propuštanjem kroz generator

3.2.2 Manipulacija vektorima u skrivenom prostoru Z

Karakteristčni vektori obeležja se mogu izdvojiti na isti način kao što je opisano kod varijacionih autoenkodera. Jedina razlika je što ne postoji direktna veza između slika iz trening skupa i vektora skrivenog prostora, pa nam labelirani atributi nisu od koristi. Zato je neophodno prvo generisati slike, pa ih zatim ručno izabrati prema zadatim atributima.

Ako je cilj generisati nasmejanog muškarca, potrebno je naći vektor nasmejane žene, oduzeti ga od vektora neutralne žene, pa zatim na rezultujući vektor dodati vektor neutralnog muškarca. Propuštanjem dobijenog vektora kroz generator generiše se nasmejani muškarac.



Slika 20: Generisanje nasmejanog muskarca manipulacijom vektora nasmejane žene, neutralne žene i neutralnog muškarca u skrivenom prostoru Z i propuštanjem kroz generator

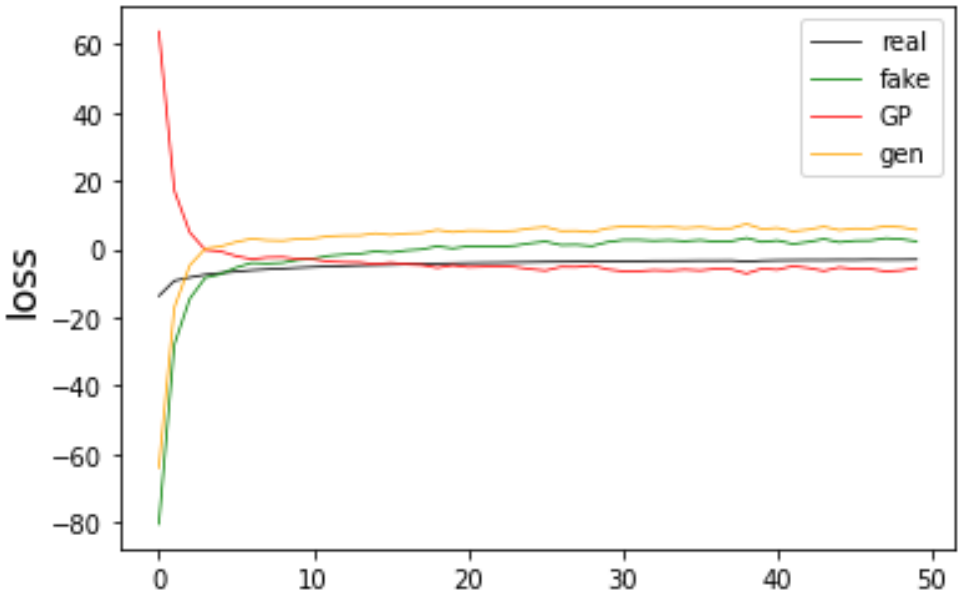
3.2.3 Prelaz iz jednog lica u drugo

Način postpenog prelaska sa jednog lica na drugo je isti kao kod varijacionog autoenkoder. Formula za "šetanje" između vektora skrivenog prostora je takođe ista: $z = z_A * (1 - \alpha) + z_B * \alpha$.

Slika 21: Postepeni prelazak iz jednog lica u drugo u 12 tačaka

3.2.4 Stabilnost treniranja modela

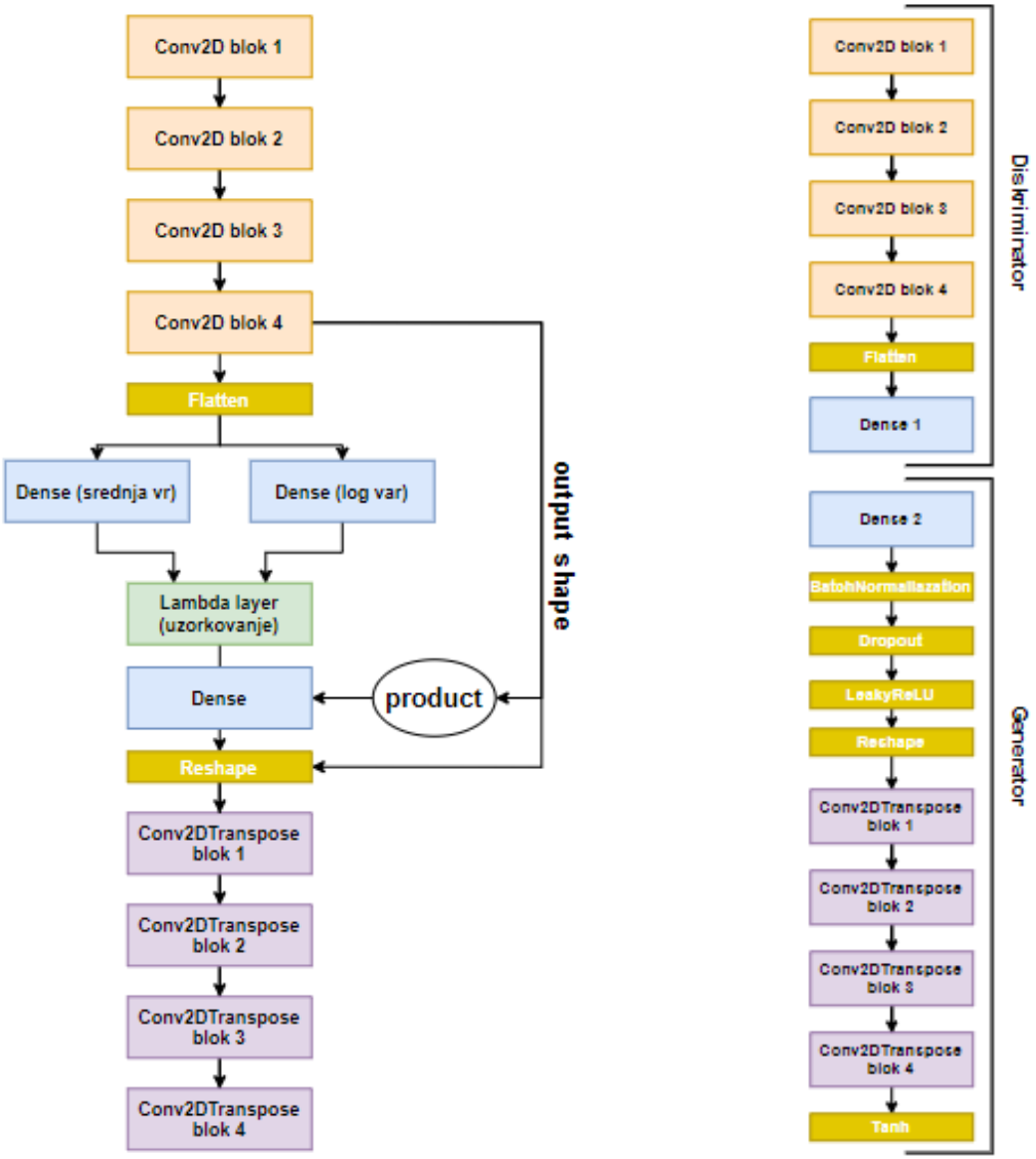
Kao što se može videti sa slike 22 treniranje je stabilno jer nema oscilacija kriterijuma.



Slika 22: Prikaz kriterijuma tokom treniranja Wassersten generativnog suprotstavljenog modela sa penalizovanjem gradijenata

4 HIPERPARAMETRI

Na slici 23 je data kompletna arhitektura Varijacionog autoenkodera i Wasserstein generativnog suprotstavljenog modela. Parametri blokovi dati su u tabelama 1 i 2. Conv2D blok se sastoji od Conv2D ⟶ BatchNormalization ⟶ Activation ⟶ Dropout slojeva. Conv2Dtranspose blok se sastoji od Conv2Dtranspose ⟶ BatchNormalization ⟶ Activation ⟶ Dropout slojeva.



Slika 23: Arhitektura Varijacionog autoenkodera (levi graf) i arhitektura Wasserstein generativnog suprotstavljenog modela sa penalizovanjem gradijenata (desni graf)

Varijacioni autoenkoder	
Hiperparametar	**Vrednost**
C	10000
dimenzije slike	128x128x3
broj epoha	200
veličina serije (eng. *batch size*)	32
veličina skrivenog prostora	200
optimizator	Adam(lr=0.0005)
broj filtera Conv2D blokova [1-4]	[32,64,64,64]
veličina kernela Conv2D blokova [1-4]	[3,3,3,3]
(eng. *strides*) Conv2D blokova [1-4]	[2,2,2,2]
(eng. *padding*) Conv2D blokova [1-4]	[“same”, “same”, “same”, “same”]
aktivacione funkcije Conv2D blokova [1-4]	[LeakyReLU, LeakyReLU, LeakyReLU, LeakyReLU]
(eng. *batch normalization*) Conv2D blokova [1-4]	[da,da,da,da]
(eng. *Dropout(rate=0.25)*) Conv2D blokova [1-4]	[da,da,da,da]
broj filtera Conv2DTranspose blokova [1-4]	[64,64,32,3]
veličine kernela Conv2DTranspose blokova [1-4]	[3,3,3,3]
(eng. *strides*) Conv2DTranspose blokova [1-4]	[2,2,2,2]
(eng. *padding*) Conv2DTranspose blokova [1-4]	[“same”, “same”, “same”, “same”]
aktivacione funkcije Conv2DTranspose blokova [1-4]	[LeakyReLU, LeakyReLU, LeakyReLU, sigmoid]
(eng. *batch normalization*) Conv2DTranspose blokova [1-4]	[da,da,da,ne]

(eng. *Dropout(rate=0.25))* Conv2DTranspose blokova [1-4]	[da,da,da,ne]

Tabela 1: Hiperparametri Varijacionog autoenkodera

Generativni suprotstavljeni model	
Hiperparametar	**Vrednost**
$n_{kritika}$	5
λ	10
dimenzije slike	64x64x3
broj epoha	60
veličina serije (eng. *batch size*)	64
veličina skrivenog prostora	100
optimizator	Adam(lr=0.0002, beta_1=0.5)
broj neurona Dense 1 bloka	1
broj neurona Dense 2 bloka	4*4*512
broj filtera Conv2D blokova [1-4]	[64,128,256,512]
veličina kernela Conv2D blokova [1-4]	[5,5,5,5]
(eng. *strides*) Conv2D blokova [1-4]	[2,2,2,2]
(eng. *padding*) Conv2D blokova [1-4]	[“same”, “same”, “same”, “same”]
aktivacione funkcije Conv2D blokova [1-4]	[LeakyReLU, LeakyReLU, LeakyReLU, LeakyReLU]
(eng. *batch normalization*) Conv2D blokova [1-4]	[ne,ne,ne,ne]
(eng. *Dropout(rate=0.25)*) Conv2D blokova [1-4]	[ne,ne,ne,ne]
broj filtera Conv2DTranspose blokova [1-4]	[256,128,64,3]
veličine kernela Conv2DTranspose blokova [1-4]	[5,5,5,5]

(eng. *strides*) Conv2DTranspose blokova [1-4]	[2,2,2,2]
(eng. *padding*) Conv2DTranspose blokova [1-4]	[“same”, “same”, “same”, “same”]
aktivacione funkcije Conv2DTranspose blokova [1-4]	[LeakyReLU, LeakyReLU, LeakyReLU, None]
(eng. *batch normalization*) Conv2DTranspose blokova [1-4]	[da,da,da,ne]
(eng. *Dropout(rate=0.25))* Conv2DTranspose blokova [1-4]	[ne,ne,ne,ne]

Tabela 2: Hiperparametri Generativnog suprotstavljenog modela sa penalizovanjem gradijenata

5 ZAKLJUČAK

Iz priloženih rezultata se može zaključiti da su oba modela naučila obeležja lica visokog nivoa i da su uspešno generisala nova lica. Varijacioni autoenkoderi proizvode mutnije slike u odnosu na generativne suprotstavljene modele, ali skriveni prostor im je dosta interpretabilniji i takođe poseduju sposobnost rekonstrukcije ulazne slike. Postepenim prelaskom jedne slike u drugu dokazano je da je skriveni prostor oba modela kompletan i kontinualan. Treniranje oba modela je stabilno, bez preteranih oscilacija u kriterijumu.

S obzirom na to da su modeli relativno novi, mesta za napredak ima dosta. Najnaprednija (eng. *state of the art*) poboljšanja ovih modela, koja proizvode slike visoke rezolucije bez ikakvih izobličenja, mogu se naći u [4,5]. Zbog velikog interesovanja naučnoistraživačke zajednice za generativne modele, očekuju su još napredniji algoritmi i modeli.

6 LITERATURA

[1] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio, ”Generative Adversarial Nets”, ”Advances in Neural Information Processing Systems 27” NIPS, Jun 2014.

[2] Martin Arjovsky, Soumith Chintala, and Leon Bottou, ”Wasserstein GAN”, arxiv, Dec 2017.

[3] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, Aaron Courville, ”Improved Training of Wasserstein GANs”, ”Advances in Neural Information Processing Systems 30” NIPS, Dec 2017.

[4] Ali Razavi, Aaron van den Oord, Oriol Vinyals, “Generating Diverse High-Fidelity Images with VQ-VAE-2“, ”Advances in Neural Information Processing Systems 32” NIPS, Jun 2019.

[5] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, Timo Aila, ”Analyzing and Improving the Image Quality of StyleGAN”, arxiv,Mar 2020.

[6] Fei Fei Li, Justin Johnson, Serena Yeung, ”Generative Models” lecture,Stanford University, Dostupno: https://www.youtube.com/watch?v=5WoItGTWV54&list=WL&index=9, Aug 2017.

[7] Joseph Rocca, ”Understanding Variatonal Autoencdoers (VAEs)”, Towards Data Science, Dostupno: https://towardsdatascience.com/understanding-variational-autoencoders vaes-f70510919f73, Sep 2019.

[8] Wickipedia, “Wasserstein metric“ Dostupno: https://en.wikipedia.org/wiki/Wasserstein_metric.

[9] Sunner Li, ”The Story about WGAN”, Medium, Dostupno: https://medium.com/@sunnerli/the-story-about-wgan-784be5acd84c, Nov 2017.

[10] Jessica Li, ”CelebFaces Atributes (celebA) Dataset”, Kaggle, Dostupno: https://www.kaggle.com/jessicali9530/celeba-dataset.

[11] Wickipedia, ”T-distributed stochastic neighbor embedding”, Dostupno: https://en.wikipedia.org/wiki/T-distributed_stochastic_neighbor_embedding

PRILOG A

Dokaz da pri optimalnom diskriminatoru kriterijumska funkcija generativnog suprotstavljenog modela zavisi od JS divergencije:

$$\mathcal{L}(\theta_g, \theta_d)) = \left(E_{x\sim p_{trening}}[\log(D_{\theta_d}(x))] + E_{x\sim p_{lažni}}[\log(1 - D_{\theta_d}(x))]\right)$$

$$= \int_x \left(p_{trening} * \log(D_{\theta_d}(x)) + p_{lažni} * \log(1 - D_{\theta_d}(x))\right)dx$$

Ako se uvede pretpostavka da je diskriminator optimalan: $\theta_d{}^* = \underbrace{\max}_{\theta_d} \mathcal{L}(\theta_g, \theta_d) \Rightarrow$

$$\frac{d\mathcal{L}(\theta_g,\theta_d)}{d\theta_d} = 0 \Rightarrow \frac{d}{d\theta_d}\left(\int_x \underbrace{\left(p_{trening} * \log(D_{\theta_d}(x)) + p_{lažni} * \log(1 - D_{\theta_d}(x))\right)}_{\leq 0, jer\ je\ 0\leq D_{\theta_d}(x)\leq 1} dx\right) =$$

$$0 \Rightarrow \frac{d}{d\theta_d}\left(p_{trening} * \log(D_{\theta_d}(x)) + p_{lažni} * \log\left(1 - D_{\theta_d}(x)\right)\right) = 0 \Rightarrow$$

$$p_{trening} * \frac{1}{D_{\theta_d}(x)} * \frac{dD_{\theta_d}(x)}{d\theta_d} - p_{lažni} * \frac{1}{1 - D_{\theta_d}(x)} * \frac{dD_{\theta_d}(x)}{d\theta_d} = 0 \Rightarrow$$

$$D_{\theta_d{}^*}(x) = \frac{p_{trening}}{p_{trening} + p_{lažni}}$$

$$\mathcal{L}(\theta_g, \theta_d{}^*)) = \int_x \left(p_{trening} * \log\frac{p_{trening}}{p_{trening} + p_{lažni}} + p_{lažni} * \log\frac{p_{lažni}}{p_{trening} + p_{lažni}}\right)dx$$

$$JS(p_{trening} \parallel p_{lažni}) =$$

$$= \frac{1}{2}KL\left(p_{trening} \parallel \frac{p_{trening} + p_{lažni}}{2}\right) + \frac{1}{2}KL\left(p_{lažni} \parallel \frac{p_{trening} + p_{lažni}}{2}\right)$$

$$= \frac{1}{2}\int_x \left(p_{trening} * \log\frac{2p_{trening}}{p_{trening} + p_{lažni}}\right)dx + \frac{1}{2}\int_x \left(p_{lažni} * \log\frac{2p_{lažzni}}{p_{trening} + p_{lažni}}\right)dx$$

$$= \frac{1}{2}\left(log4 + \int_x \left(p_{trening} * \log\frac{p_{trening}}{p_{trening} + p_{lažni}} + p_{lažni} * \log\frac{p_{lažni}}{p_{trening} + p_{lažni}}\right)dx\right)$$

$$= \frac{1}{2}\left(log4 + \mathcal{L}(\theta_g, \theta_d{}^*)\right)) \Rightarrow \boxed{\mathcal{L}(\theta_g, \theta_d{}^*) = 2 * JS(p_{trening}(x) \parallel p_{lažni}(x)) - 2 * log2}$$