

Elektronski fakultet u Nišu

Katedra za Računarstvo i informatiku

Neural Style Transfer

Soft computing

Student: Nemanja 919

Mentor: Aleksandar Milosavljević

Uvod	3
Tip Generativnog Neural Style Transfer-a	4
Funkcija sadržaja (content loss)	5
Funkcija stila (style loss)	6
Gram matrica	6
Računanje loss funkcije stila na osnovu gram matrice	7
Generisanje slike pomoću neural style transfer-a	8
Odabir konvolucione mreže	8
Odabir slojeva konvolucione neuronske mreže	9
3. Implementacija projekta	10
4. Rezultati	12
Zaključak	14
Reference	15

1. Uvod

Neural style transfer je tehnika optimizacije koja koristi dve slike - **slike sadržaja** i **slike stila** (poput umetničkog dela poznatog slikara) - i kombinuje ih tako da izlazna slika izgleda kao slika sadržaja, ali „slikana“ u stilu referentne slike stila. Odnosno primeni stil slike jedne slike na sadržaju druge. Postoji veći broj tehnika kako se to može postići, od same tehnike dosta zavisi kvalitet dobijenog rezultata kao i brzina generisanja izlazne slike na osnovu druge dve. Ovakva primena mašinskog učenja je jako interesantna jer su rezultati ja

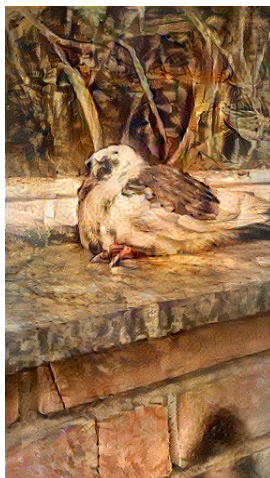
Ova implementacija je zasnovana na optimizaciji izlazne slike kako bi se podudarala sa statistikom sadržaja slike sadržaja i statistikom stila referentne slike stila. Ove statistike se izdvajaju iz slika pomoću konvolucione mreže.

2. Tip Generativnog Neural Style Transfer-a

Osnovna ideja ovog tipa generativnog Neural Style Transfera je da se iskoriste feature mape pretrenirane CNN (konvolucione neuronske mreže) kako bi se dobile vektorske reprezentacije odgovarajućih elemenata slika i na njih primenile funkcije koje respektivno izvlače numeričku reprezentaciju prisutnosti sadržaja na slici, kao i prisutnosti stila neke slike.



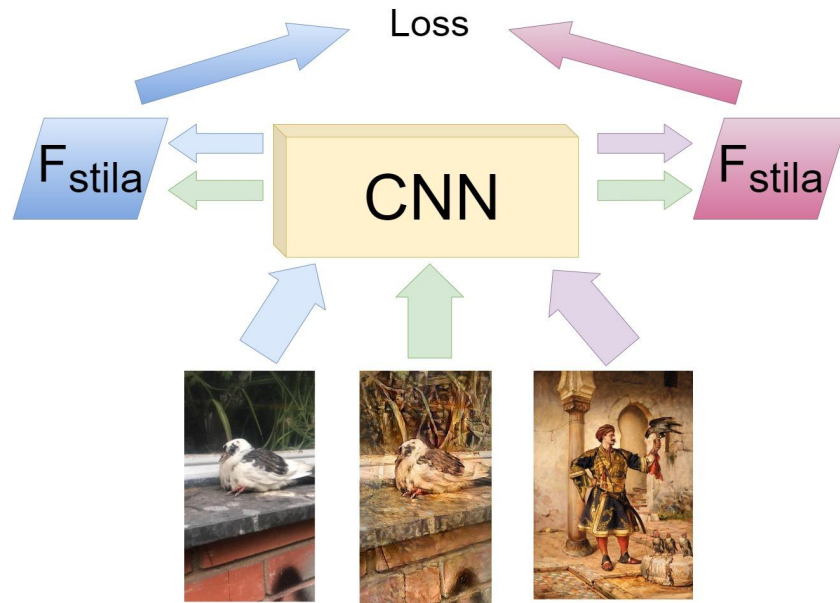
Slika1. Primer slike stila i sadržaja



Slika2. Primer generisane slike

Odnosno koriste se izlazi odgovarajućih filtera konvolucione neuronske mreže i oni se propuštaju kroz odgovarajuće loss funkcije u kojima se porede izvorne slike sa generisanom slikom. Time se zapravo iterativno posmatra prisutnost sadržaja i stila na generisanoj slici, i onda se na osnovu formule loss-a pravi balans sadržaja i stila. Zato je potrebno da postoje 2 slike, takozvana slika stila (eng. Style image) i slika sadržaja (eng. Content Image). Poredjenjem slike stila i generisane slike pomoću funkcije greške stila se dobija “greška generisanog sadržaja u stilu”, dok se poredjenjem generisane slike i slike sadržaja u funkciji greške sadržaja

dobija “kolika je greška generisanog sadržaja”. Balansiranje ove dve greške se dobija matematička kombinacija te dve slike.



Slika 3. Vizualizacija procesa računanja loss-a

Funkcija sadržaja (content loss)

Ova funkcija je prosta i nije potrebno puno intuicije da se razume kako radi. Samo je potrebno da se uzme euklidska udaljenost matičnih reprezentacija tih slika. Potrebno je propustiti slike kroz konvolucionu mrežu i uzeti izlaze iz određenih slojeva. Nakon toga izračunati euklidisk razlike između slika u svakom pred određenom sloju. Matematički možemo zapisati na sledeći način, pri čemu je l - odabrani sloj, x - generisana slika, p - slika sadržaja, F - izlaz konvolucione mreže za generisanu sliku, P - izlaz konvolucione mreže za sliku p , i (j) - indeksi piksela u slici (matrici).

$$\mathcal{L}_{content}(\vec{p}, \vec{x}, l) = \frac{1}{2} \sum_{i,j} (F_{ij}^l - P_{ij}^l)^2 .$$

Slika 4. Funkcija sadržaja (content loss)

Funkcija stila (style loss)

Računanje greške stila je komplikovanije od računanja greške sadržaja. Dok se greška sadržaja računa samo kao euklidska razlika između izlaza mreže, za stil je prvo potrebno generisati gram matricu koja nam matrično reprezentuje stil izlaza nekog konvolucionog sloja.

Gram matrica

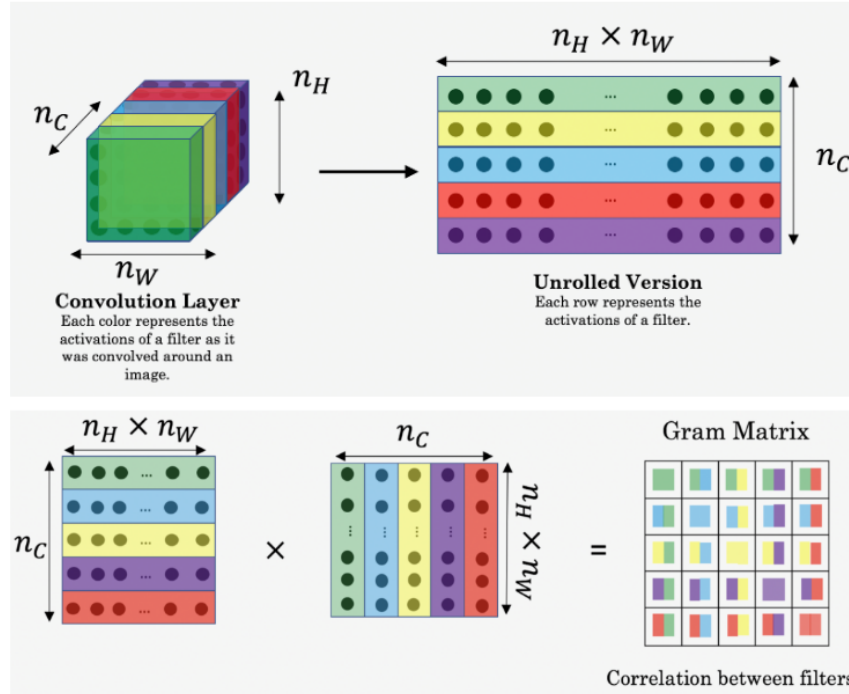
Stil slike se može opisati korelacijama na različitim feature mapama. Izračunavanjem Gram matrice koja predstavlja kombinaciju svih filtera mreže. Ona se računa tako što se uzima tenzorni proizvod vektora sa sobom na svakoj poziciji i uzimajući srednju vrednost tenzornog proizvoda po svim lokacijama. Tako se pomoću gram matrice mogu izvući informacije o stilu iz izlaznog sloja, i sama matrica se može koristiti za poredjenje stilova dveju slika. Ova Gramova matrica može se izračunati za određeni sloj kao:

Ako uzmemo da je G gram matrica, F - vektorizovana feature mapa l -tog sloja konvolucione neuronske mreže, a prolazi se kroz sumu broja kanala.

$$G_{ij}^l = \sum_k F_{ik}^l F_{jk}^l.$$

Slika 5. Funkcija gram matrice

Tako se dobija matrična reprezentacija stila neke slike, a izlaz je veličine $k \times k$ pri čemu je k - broj filtera u odabranom sloju pretrenirane konvolucione mreže. Tako se mogu izračunati gram matrice većeg broja slika onda se mogu porediti te matrice, čime se suštinski porede stilovi u odgovarajućim slojevima konvolucione neuronske mreže.



Slika 6. Vizuelizacija računanja gram matrice pomoću feature matrica

Računanje loss funkcije stila na osnovu gram matrice

Prethodni podnaslov se može iskoristiti kako bi izračunali grešku generisanog stila neke slike. Naime možemo uzeti generisanu sliku i sliku stila, propustiti ih kroz konvolucionu mrežu i izvući feature mape. Nakon toga možemo na osnovu tih feature mapa da izračunamo gram matrice, koje bi nam poslužile kao matrična reprezentacija stila slike.

Ako uzmemo da je G gram matrica generisane slike a A gram matrica slike stila i one nam predstavljaju matričnu reprezentaciju stila. Možemo da definišemo loss koji poredi te dve matrice i kaže nam grešku stila za odgovarajuće l slojeve konvolucione matrice. Pri tome N, M su veličine feature mape (feature mapa je $N \times M$ veličine). Kad dobijemo loss po sloju možemo da napravimo težinsku sumu $w \cdot E$ grašaka po slojevima, čime bi težinski uzimali greške po slojevima, odnosno mogli bi da definišemo koliko koji slej je bitan za generalnu grešku stila.

$$E_l = \frac{1}{4N_l^2 M_l^2} \sum_{i,j} (G_{ij}^l - A_{ij}^l)^2$$

$$\mathcal{L}_{style}(\vec{a}, \vec{x}) = \sum_{l=0}^L w_l E_l$$

Slika 6: Funkcija greške stila

Generisanje slike pomoću neural style transfer-a

Kao što je već napomenuto generisanje slike pomoću ovog algoritma za Neural Style Transfer je iterativni proces u kome se inkrementalno generisana slika unapređuje, odnosno smanju joj se greška stila i sadržaja.

Prvo je potrebno odabrati neku konvolucionu mrežu koja je pretrenirana na sličnom setu slika koji bi se koristio za neural style transfer, zatim se odaberu dve slike (sliku stila, i sliku sadržaja). Zatim se odluči koji slojevi će se koristiti prilikom generisanja neural style transfera, logika iza odabira slojeva će biti kasnije objašnjena. Slika sadržaja se uzima kao inicijalna generisana slika i ona se “unapređuje”, odnosno na njoj se pravi balans stila i sadržaja druge sve slike. Nakon toga sledi iterativni proces generisanja slike.

1. Postoje 3 slike C-slika sadržaja, G-generisana slika i S-slika stila.
2. Sve tri slike se propuste kroz konvolucionu neuronsku mrežu i za odgovarajuće slike se izvlače feature mape iz odgovarajućih slojeva.
3. Za feature mape slike stila i generisane slike se računa gram matrica kako bi mogla da se odredi sličnost stila.
4. Računaju se graške generisane slike u odnosu na sliku stila i sliku sadržaja
5. Na osnovu dobijene vrednosti greške, izvršava se gradient descent algoritam na generisanoj slici.

Odabir konvolucione mreže

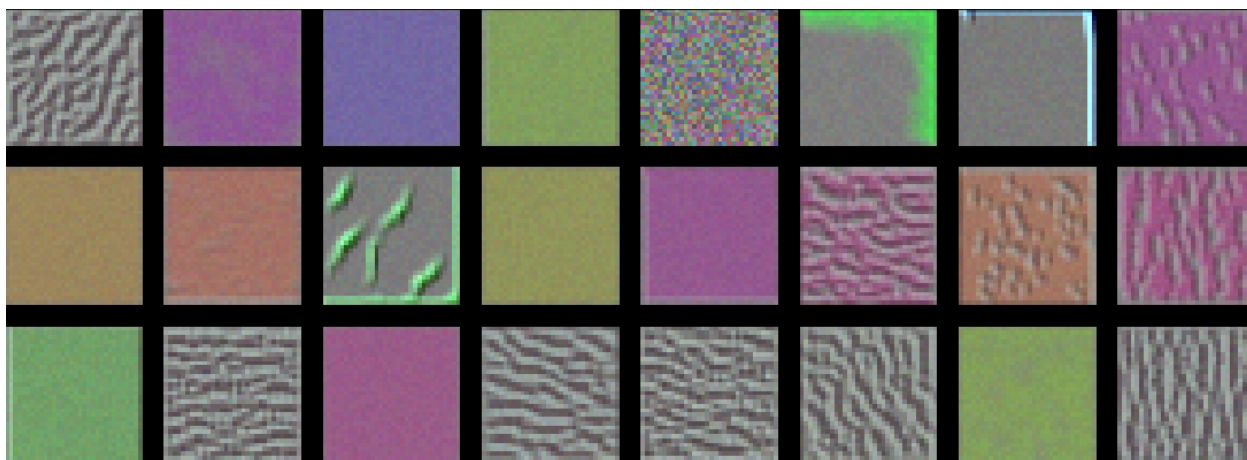
Ne postoje neka posebna pravila koja možemo definisati za odabir konvolucione mreže. Nešto što je neophodno je da ta mreža bude pretrenirana na većem setu podataka. Još nešto što se treba uzeti u obzir jeste broj slojeva. Za funkciju stila i sadržaja je bitno odabrati odgovarajuće slojeve (kako se biraju je opisano u sledećem naslovu), pa samim tim je potrebno uzeti u obzir da konvoluciona mreža treba da ima takve slojeve koje možemo uzeti za generisanje slike. Samim tim mreža treba da bude malo dublja (ne sme pretrano plitka da bude), a iz poznavanja mehanizma konvolucije možemo da zaključimo da je dobro da mreža bude pretrenirana na sličnom setu slika kakve će se koristiti kao slike sadržaja. Potrebno je da konvoluciona neuronska mreža razume sadržaj slike kako bi mogla da ga sačuva, ukoliko mreža nije pretrenirana za određene oblike verovatno se može desiti da ih promeni u generisanoj slici, ili da na njima nastanu nekakvi defekti koji bi mogli da umanje kvalitet dobijene generisane slike.

Odabir slojeva konvolucione neuronske mreže

Postoje različiti alati i tehnike za vizuelizaciju izlaza, aktivacija i heatmap-a konvolucionih neuronskih mreža. Možemo iskoristiti takve alate da malo bolje razumemo kako i zašto se odgovarajući slojevi biraju za generisanje pomoću neural style transfera.

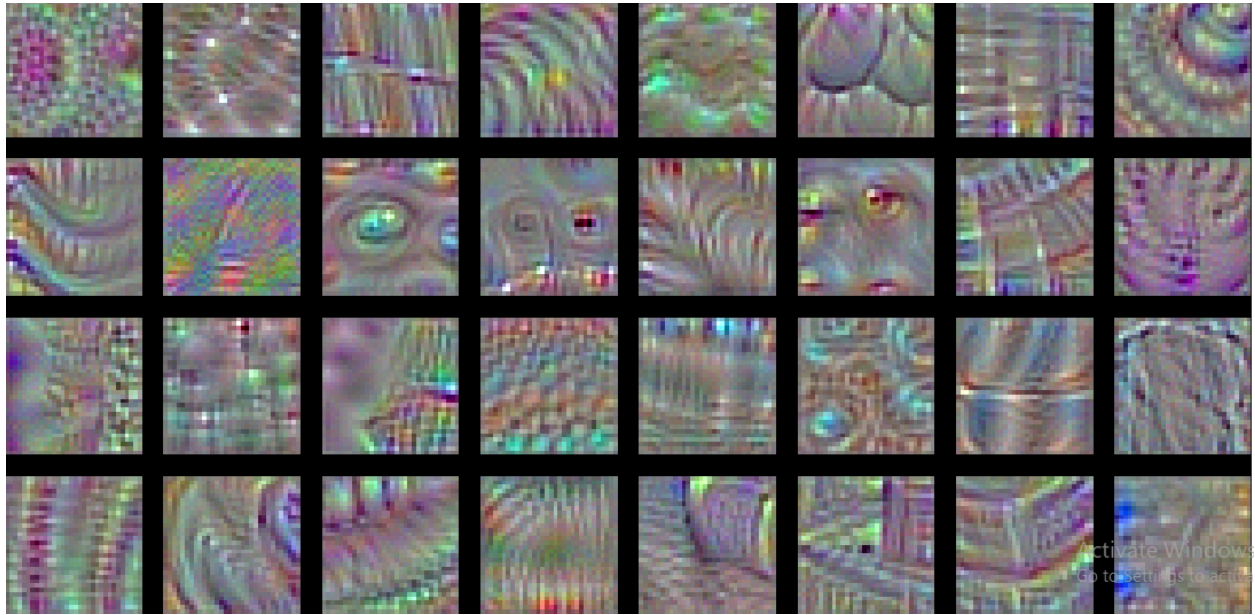
Najbitnija razlika koju moramo primetiti prilikom odabira slojeva je upravo čemu će da se koriste ti slojevi. Slojevi koji se koriste za proračunavanje sličnosti generisanog stila sa stilom neke slike, njima je bitno da zapravo u nekom smislu reprezentuju stil slike. Odnosno slojevi za stil funkciju treba da u konvolucionalnoj mreži budu odgovorni za neki apstraktni stil, ili nekih primitiva koje su korišćene u slikanju. Slojevi koji se koriste u izračunavanju greke razlika sadržaja slike je potrebno da u konvolucionalnoj mreži služe za prepoznavanje nekog sadržaja na slici.

Ako iskoristimo vizuelizaciju filtera konvolucione neuronske mreže, generišamo slike koje najviše aktiviraju odgovarajuće filtere. Time možemo da vizualizujemo koje vrste oblika/ slike odgovarajući slojevi prepoznaju, tačnije za koje oblike se najviše aktiviraju odgovarajući filteri.



Slika 7. Vizuelizacija filtera prvih slojeva konvolucione mreže

Prva slika prikazuje aktivacije prvih slojeva konvolucione mreže. Ono što možemo da zapazimo jeste da su u pitanju prosti oblici, neke vrste linija, krivih. Ti prvi slojevi služe za prepoznavanje oblika, na primer horizontalnih, vertikalnih linija. Kako idemo u dubinu onda postaju apstraktnije strukture. Polako se prelazi sa nekog filtera slike u internu reprezentaciju konvolucione mreže strukture slike. Tačnije kako konvoluciona mreža razume celu sliku. Vidi se jasna razlika između slika, prve slike imaju neke osnovne primitive, dok se na drugim slikama vide apstraktni oblici.



Slika 8. Vizuelizacija krajnjih filtera konvolucione neuronske mreže

3. Implementacija projekta

Projekat je implementiran u Python-u pomoću tensorflow 2.0 okruženja (biblioteke) uz nju su korišćene i biblioteke: Keras (za definisanje modela CNN mreže), tqdm(za vizuelizaciju procesa generisanja modela), open cv (za vizuelni prikaz procesa generisanja kao i za snimanje generisane matrice kao .jpg slike)

Tensorflow 2.x je trenutno aktivna biblioteka/ framework koji je napravio google i u samoj biblioteci je ugradjen keras kao high level api za tensorflow.

Sama implementacija je u okviru jednog PyCharm python projekta i može se pozivati kao skripta kojoj se daju putanje do slike stila i slike sadžaja, kao i putanja do direktorijuma gde se snima generisana slika.

Samo generisanje slike traje i do par minuta, sve u zavisnosti od broja epoh-a koji je definisan za generisanje, generalno nakon odredjenog broja iteracija generisana slika prestane da dobija neke veće izmene i polako greška generisane slike ulazi u neki lokalni optimum (minimum). Tako da je moguće i prekinuti generisanje slike ukoliko je korisnik vizuelno zadovoljan rezultatima.

Projekat je realizovan kao main.py Python skripta koja se može pozvati iz terminala sa dodatnim parametrima. Postoje sledeći parametri:

1. path_content - putanja do slike koja se koristi kao slika saržaja
2. path_style - putanja do slike koja će se koristiti kao slika stila
3. path_output - putanja do direktorijuma u kojem će se snimati generisane slike

4. title - naziv generisane slike (pod tim imenom će se snimati generisane slike)
5. save_epoch - boolean vrednost koja služi kao flag koji određuje jel će se snimanje generisane slike raditi nakon svake epohe ili na kraju.

Sam proces generisanja slike je vrapovan u okviru klase NeuralStyleModel koja ima sve potrebne funkcije za generisanje neural style slike. Sam mogel koji se koristi prilikom generisanja slike je vrapovan u okviru keras model klase StyleContentModel koja u sebi sadrži VGG19 mrežu i potrebne funkcije za procesiranje neural style transfera.

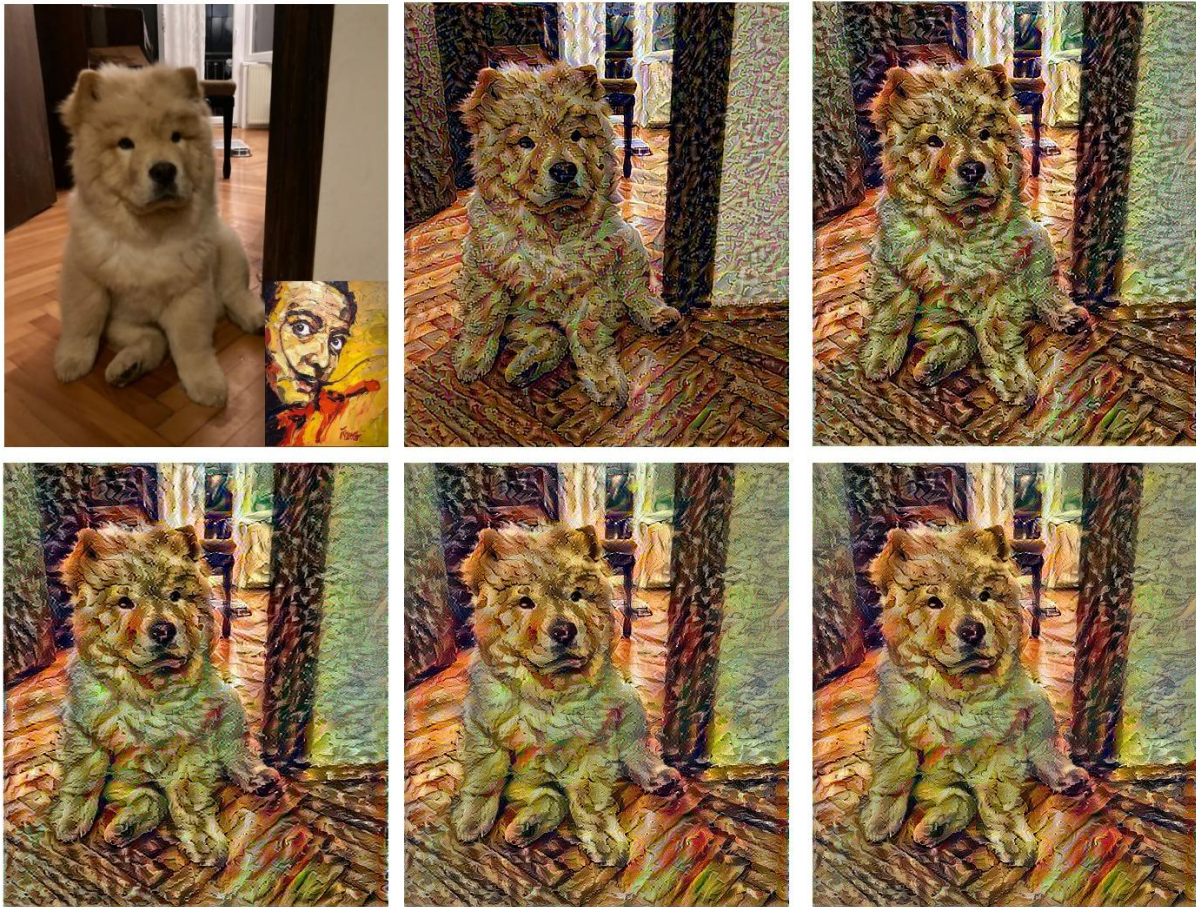
Odabir slojeva konvolucione neuronske mreže za neural style transfer:

Stil slojevi - block1_conv1, block2_conv1, block3_conv1, block4_conv1, block5_conv1

Odabran je neki srednji tok slojeva kako bi se uzeli prostiji i složeniji detalji stila slike. Tu se onda vide proste primitive, potezi četkica, kao i složene ideje stila samog slikara

Sadržaj sloj - block5_conv2

Odabran je jedan malo dublji sloje jer konvoluciona mreža u prvim slojevima priprema sliku a u kasnijim pravi neku apstrakciju slike koja joj omogućava da prepozna šta je na slici. Tako da je potrebno da ta neka apstrakcija koja zapravo predstavlja objekat na slici koju mreža koristi za prepoznavanje u stvari bude slična. Samim tim ukoliko se prepozna neki objekat naslici A potrebno je da i on bude prepoznat na slici B.



Slika 9. Iterativni proces generisanja neural style slike

4. Rezultati

Isprobavao sam neural style algoritam na većem broju slika sadržaja i slika stila. Neki moji rezultati se mogu videti na narednoj slici, pored toga postoji nekoliko mojih zapažanja koja se tiču ove implementacije neural style transfera.

1. Sam proces generisanja može da traje od desetak sekundi do minut dva, u zavisnosti od broja itreacija generisanja slike.
2. Različite slike mogu da zahtevaju dosta različit broj iteracija da se dobije lokalni minimum greške generisane slike.
3. Nisam zapazio da je overfitovanje prilikom generisanja slike neki značajan problem. U nekim situacijama su mi vizuelno prijale generisane slike koje nisu bile poslednje. Ali razlika nije značajna, u nekim situacijama je jako teško primetljiva.

4. Najbolje je uzeti sliku sa jednim centralnim objektom, slike sa većim brojem aktera mogu biti mutne i sa raznim defektima.
5. Slike lica ukoliko se puštaju kroz NST mogu biti malo zamućene i iskrivljene.
6. Veći broj slika ima čudne pozadinske defekte sadržaja, pozadina sadržaja slike uglavnom ne bude prepoznata kao bitan sadržaj pa u tom delu preovlada stil slike.



Slika 10. Primeri generisanja slike pomoću neural style transfer algoritama

Zaključak

Neural style transfer je jako zanimljiva primena machine learning algoritama jer u sebi ima tu neku umetničku crtu. Uz to je jako dobar primer za učenje konvolucionih neuronskih mreža jer je tema jako zanimljiva a da bi se razumela mora se razumeti sam način funkcionisanja konvolucionih neuronskih mreža. Mora se shvatiti kako se koriste feature mape i kako sama mreža razume slike, kako ih postepeno filtrira i pravi neku svoju strukturu koju nakon toga koristi za prepoznavanje sadržaja slike. Sam algoritam ovog generisanja nije toliko složen i može se razumeti. Ovaj algoritam nije najoptimalniji, jer postoji veći broj metoda koje daju bolje rešenja koja su uz to i znatno brža. Ovaj pristup zahteva da se gradient descent izvršava iterativno za svaki proces generisanja slike što može dovesti do znatnog usporenja. Prilikom istraživanja problema sam naišao na veći broj drugih pristupa koji daju i vizuelno više zadovoljavajuće rezultate tako da je ovaj pristup na neki način zastareo, međjutim meni je bio zanimljiv jer se njime dosta duboko zalazi u konvolucine neuronske mreže a sam rezultat ispada interesantan.

Reference

1. <https://medium.com/tensorflow/neural-style-transfer-creating-art-with-deep-learning-using-tf-keras-and-eager-execution-7d541ac31398><https://medium.com/tensorflow/neural-style-transfer-creating-art-with-deep-learning-using-tf-keras-and-eager-execution-7d541ac31398>
2. https://www.tensorflow.org/tutorials/generative/style_transfer
3. Gatys, Leon A., Alexander S. Ecker, and Matthias Bethge. "A neural algorithm of artistic style." *arXiv preprint arXiv:1508.06576* (2015).