

UNIVERZA V LJUBLJANI
FAKULTETA ZA RAČUNALNIŠTVO IN INFORMATIKO

Primož Godec

**Barvanje črnobelih slik z globokimi
modeli**

MAGISTRSKO DELO
MAGISTRSKI PROGRAM DRUGE STOPNJE
RAČUNALNIŠTVO IN INFORMATIKA

MENTOR: prof. dr. Blaž Zupan

Ljubljana, 2017

AVTORSKE PRAVICE. Rezultati magistrskega dela so intelektualna lastnina avtorja in Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavljanje ali izkoriščanje rezultatov magistrskega dela je potrebno pisno soglasje avtorja, Fakultete za računalništvo in informatiko ter mentorja.

ZAHVALA

Na tem mestu zapišite, komu se zahvaljujete za izdelavo magistrske naloge. V zahvali se poleg mentorja spodobi omeniti vse, ki so s svojo pomočjo prispevali k nastanku vašega izdelka.

Primož Godec, 2017

Vsem rožicam tega sveta.

*"In nature, light creates the color.
In the picture, color creates the light."*

— Hans Hofmann

Contents

Povzetek

Abstract

1	Uvod	1
2	Pregled področja	5
2.1	Obstoječe metode za barvanje črno-belih slik	5
2.2	Globoke nevronske mreže	7
2.3	Predstavitev slikovnih podatkov in barvni prostori	10
3	Barvanje črno-belih slik z globokimi nevronskimi mrežami	15
3.1	Arhitekture	15
3.2	Pristopi z regresijo	19
3.3	Pristopi s klasifikacijo	22
3.4	Cenilne funkcije	24
3.5	Postopek učenja	26
3.6	Barvanje večjih slik	29
3.7	Podatki	31
3.8	Računanje napake	32
4	Rezultati in diskusija	35
4.1	Primerjava metod na manjši učni množici	35
4.2	Primerjava metod na večji učni množici	41
4.3	Barvanje večjih slik	41

CONTENTS

5 Zaključek	45
A Spearmanova korelacija rangov med metodami	47
B Podrobnosti arhitektur	49

Povzetek

Naslov: Barvanje črnobelih slik z globokimi modeli

Barvna fotografija je prišla v vsakdanjo uporabo šele v zadnjih 50 letih, zato imajo naši stari starši še vedno mnogo črno-belih fotografij, katere radi pogledajo in obujajo spomine. Bolj prijetno je gledati na fotografije, ki imajo prisotnih saj nekaj barv. Te obenem tudi bolj naravno prikažejo objekte in ljudi na sliki. To je razlog za razvoj različnih pristopov za barvanje črno-belih fotografij in videov.

V tem delu predstavljamo nove avtomatske pristope za barvanje slik po delih, ki bazirajo na konvolucijskih nevronskeih mrežah. Te pristopi nam omogočajo barvanje slik želene velikosti skoraj z enako natančnostjo kot tiste na katerih je bila mreža učena. V delu predstavljamo tudi primerjavo naših pristopov z pristopih iz sorodnih del.

Ključne besede

Umetna inteligenco, odkrivanje znanj iz podatkov, globoko učenje, nevronske mreže

Abstract

Title: Deep models for image coloring

Because the color photography came in everyday use in last fifty years our grandparents are still owning many black and white photographs which they like to look at to remember the old times. It is more pleasant to look at photos with some colors than a black and white one, they also look more natural. It encourages researchers to develop approaches for black and white photographs and video colorization.

In this thesis we present new approaches for automatic black and white photo colorization based on convolutional neural networks which are working on parts of images instead of taking into account full image. That approaches allows us to color images of different size to those used for training with almost equal accuracy. We also present comparison between our own approaches and those presented in related works.

Keywords

Artificial intelligence, data mining, deep learning, neural networks

Poglavlje 1

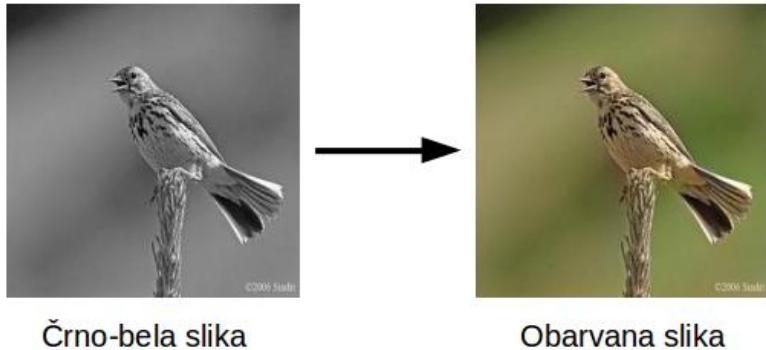
Uvod

Čeprav se so prvo barvno fotografijo naredili že leta 1886 [1], se je barvna fotografija v vsakdanji uporabi uveljavila šele mnogo kasneje. Tako imajo naši stari starši še vedno veliko črno belih fotografij. Ker te prikazujejo realnost povsem drugače, bi jih radi obarvali. Ali je to sploh mogoče?

Barvanja črno-belih fotografij so se lotevali že v devetnajstem stoletju, ko so to počeli ročno. V sedemdesetih letih prejšnjega stoletja so se pojavili prvi pristopi z računalnikom, ki so še vedno zahtevali nekaj uporabnikovega sodelovanja. Kasneje so se pristopi izboljševali in postali vedno bolj avtomatski.

Algoritmi za barvanje črno-belih slik dobijo kot vhod črno-belo sliko, ki ji dodajo barvo, kot je prikazano na sliki 1.1. Pristopi za barvanje črno-belih slik se uporabljam na več področjih: barvanje starih slik, barvanje črno-belih filmov in v pomoč pri umetnosti. V preteklosti so bili pristopi za barvanje slik pol-avtomatski, danes pa jih zamenjuje pristopi, ki obarvajo sliko popolnoma samostojno. Zadnje raziskujemo v tem magistrskem delu.

Za človeka je barvanje črno-belih slik, ki so prikazane na sliki 1.2, enostavna naloga. Z vsakdanjim opazovanjem sveta se je človek naučil, da je nebo modro z belimi oblaki, drevesa so zelena in cesta je siva. Za objekte, ki nimajo enolično določene barve ljudje lahko ugibamo kakšne barve naj bi bili. Pri tem opravilu je potrebno veliko razumevanja, saj iz sivinskih slik ni možno direktno razbrati barv. Pri nastanku teh se dve tretjini informacij



Slika 1.1: Pristop za vhod vzame črno-belo (sivinsko) sliko, preko nivojev nevronske mreže določi barvne komponente in na izhodu vrne obarvano sliko.

izgubi, kar dve od treh dimenzij.

Problem postane bolj kompleksen, ko ga želimo rešiti na avtomatski način z računalnikom. Pri tem nam je v pomoč dejstvo, da je možno iz tekstur objektov te prepoznati in jim na ta način določiti njihovo barvo. Pri objektih, ki nimajo enolično določenih barv (na primer avtomobili, stavbe in knjige) je izziv mnogo težji. Pri tem nam delo olajša dejstvo, da ne želimo, da slika zgleda enaka originalni ampak, da ima kar se da naravni izgled. Nihče ne bo vedel, da je avtomobil, ki smo ga pobarvali modro bil v resnici rdeče barve.

Za barvanje slik smo izbrali pristope, ki uporabljajo nevronske mreže. Te delujejo podobno, kot človeški možgani. Na začetku jih naučimo tako, da jim podamo čim več primerov barvanja slik. Kasneje nevronska mreža naučeno znanje uporabi, da obarva slike. Mreži podamo sivinsko sliko, ki je kar L kanal barvnega prostora *CIE Lab*, vrne pa nam a in b kanal v istem barvnem prostoru. Za učenje modela potrebujemo veliko količino črno-belih slik z referenčno barvno sliko, kar je praktično zastonj na voljo na spletu. Za treniranje lahko vzamemo katerokoli sliko, jo pretvorimo v barvni prostor *CIE Lab*, kjer L kanal predstavlja sivinsko sliko.



Slika 1.2: Primeri barvanja črno-belih slik. Barvanje je bilo izvedeno s pristopi, ki so bili razviti v okviru magistrskega dela. Za vsako sliko je prikazana črno-bela slika, ki je bila vhod v algoritmom in obarvana slika - izhod algoritma.

V tej magistrski nalogi rešujemo problem barvanja črno-belih slik in videov z večimi pristopi. Za začetek smo implementirali več algoritmov za barvanje črno-belih slik. Rezultate smo med seboj primerjali z računanjem razlike med obarvano in originalno sliko, kar je napaka barvanja. Rezultate smo primerjali s tremi pristopi iz sorodnih del. Ker obstaja veliko objektov, ki nimajo enolične barve in naš namen ni doseči enakega barvanja, vendar takega, ki da naravne rezultate, smo barvanje slik ocenjevali s pomočjo uporabnikov. V spletni anketi smo uporabnike spraševali katera slika je bolj naravno obarvana (originalna ali slika obarvana z algoritmom).

Kasneje smo se odločili poskusiti tudi barvanja vida. Za barvanje smo arhitekturo nevronske mreže, ki je najbolje delovala na slikah prilagodili za video.

Na začetku si bom v pregledu področja pogledali sorodna dela za barvanje slik, nekaj ozadja o podatkih in globokih nevronskih mrežah. V poglavju 3 so podrobno opisane arhitekture nevronskih mrež in različni pristopi. Opisano je učenje nevronskih mrež, podatki in način evalvacije. V poglavju 4 smo primerjali pristope razvite za namen tega magistrskega dela z pristopi iz sorodnih del in si pogledali kateri pristopi in slike najbolj izstopajo. Preizkusili smo kateri pristop deluje najbolje na slikah večjih velikosti.

Poglavlje 2

Pregled področja

V tem poglavju si bomo pogledali obstoječe pristope za barvanje črno-belih slik, si pogledali področje globokih nevronskih mrež ter nekaj o predstavitev slikovnih podatkov in barvnih prostorih.

2.1 Obstojče metode za barvanje črno-belih slik

Pristope za barvanje črno-belih slik delimo v dve večji skupini. Prva zahteva interakcijo uporabnika med postopkom barvanja in se je več uporabljala predhodno, pri drugi pa barvanje poteka popolnoma avtomatsko.

2.1.1 Pristopi, ki zahtevajo interakcijo uporabnika

To skupino pristopov delimo na tehnike, ki temeljijo na uporabnikovem barvanju manjših delov slik (ang. *scribble based*) [2, 3] in tiste, ki temeljijo na primerih (ang. *example based*) [4, 5, 6]. Pri prvih v osnovi uporabnik določi barvo nekaj točk v sliki, te pa algoritem avtomatsko razširi preko cele slike. Kvaliteta barvanja je odvisna od zahtevnosti slike in števila točk, ki jih je uporabnik označil. Pri barvanju na primerih uporabnik izbere referenčno sliko podobno tisti, ki jo želimo obarvati, algoritem pa lastnosti izbrane slike

razširi na drugo sliko ali množico slik. Kvaliteta barvanja je odvisna od tega v kolikšni meri je referenčna slika podobna tisti, ki jo barvamo. Tehnika barvanja s primeri se uporablja za barvanje videov, saj je v tem primeru potrebno ročno pobarvati na primer vsako stoto sliko, na ostale pa algoritom sam razširi lastnosti ročno barvane slike.

2.1.2 Popolnoma avtomatski pristopi

V magistrskem delu se osredotočamo na avtomatske pristope barvanja. Ti pristopi samostojno brez uporabnikovega posredovanja, obarvajo celotno sliko. Prvi dve metodi, ki sta bili predlagani na tem področju, temeljita na značilkah (ang. *features*) pridobljenih iz slike. Tukaj gre predvsem za značilke, ki opisujejo intenzitet posamezne barve in značilke, ki opisujejo robove v sliki. Prva metoda uporablja za barvanje nevronsko mrežo [7], ki pa vsebuje zgolj polnopovezane nivoje, druga pa za barvanje uporabi metodo naključnih gozdov (ang. *random forest*) [8].

Novejši pristopi barvanja črnobelih slik tipično temeljijo na konvolucijskih nevronskih mrežah, ki imajo to lastnost, da v vsakem nivoju same odkrijejo značilke, ki so pomembne za čim bolj kvalitetno barvanje. Prva tovrstna rešitev [9] gradi mrežo na podlagi že zgrajene šestnajst-nivojske mreže VGG-16, ki so jo razvili na univerzi v Oxfordu [10]. Rešitev uporablja evklidsko cenilno funkcijo in barvni prostor YUV. Slabost te rešitve je, da izhodne barvne slike niso dovolj nasičene in imajo v veliki meri prisotnih več rjavih odtenkov.

V zadnjem času predlagane rešitve popravijo problem nenasičenosti z uporabo softmax funkcije v zadnjem nivoju nevronске mreže, kar pomeni, da so problem spremenili iz regresijskega v klasifikacijskega. Zang in sod. [11] uporabijo konvolucijsko nevronsko mrežo z več nivoji in aktivacijskimi funkcijami ReLU. Posebnost te mreže je cenilna funkcija. Uporablja križno entropijo, ki pa je v tem primeru izvedena na primerjavi barv posameznih delov slike glede na barvni prostor, ki je kvantiziran. Napake so pomnožene z utežjo, ki določa pogostost barve. Bolj redke barve so obtežene tako, da prispevajo

večji delež k napaki, ki jo izračuna cenilna funkcija. S tem so avtorji izboljšali rezultate, tako da se bolj pogosto pojavljajo tudi močnejši odtenki (tisti z višjimi vrednostmi v prostoru a^*b^* , ki so bili prej redkeje zastopani zaradi bolj pogostega pojavljanja nežnejših barv v slikah (barve bližje vrednostim $(0, 0)$ v a^*b^* prostoru). Pogostost je bila izračunana z analizo vseh slik v podatkovni zbirki Imagenet [12]. Uporabljajo barvni prostor $L^*a^*b^*$.

Larsson in sod. [13] za osnovo uporabijo mrežo VGG-16, iz katere vzamejo tenzorje vsakega nivoja, ki jim povečajo prostorsko dimenzijo, tako da se ujemajo in združijo v enotno matriko. Sledi še en polno-povezan nivo na nivoju točk v sliki. Rezultat klasifikacije je histogram z verjetnostmi posameznega odtenka za vsako točko v sliki. Uporabljajo barvni prostor HSV, ki ga prilagodijo zaradi nestabilnosti v eni od točk. Cenilna funkcija, ki jo uporabljajo je KL-divergenca, ki primerja izhodni histogram z v histogram pretvorjeno originalno sliko.

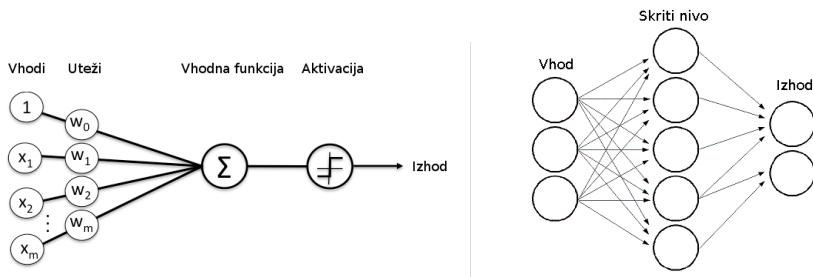
Iizuka in sod. [14] uporabijo nevronske mreže sestavljene iz dveh delov. Prvi del poskrbi za napovedovanje vsebine slike, ki se potem združi z glavnim delom in izboljša natančnost barvanja. Uporabili so križno entropijo (ang. *cross entropy*) v kombinaciji s cenilno funkcijo *povprečna kvadratna napaka* (ang. *Mean squared error*) in barvni prostor $L^*a^*b^*$. Za razliko od prejšnjih dveh metod zadnja ne napoveduje histograma na podlagi kvantiziranega prostora ampak direktno a^* in b^* vrednost, kar pomeni, da ne uporablja klasifikacije ampak regresijo.

2.2 Globoke nevronske mreže

Globoke nevronske mreže so algoritmi, ki so zgrajeni na podlagi opazovanja strukture možganov. Uporabljajo se za klasifikacijo, regresijo, gručenje in napovedovalno analizo. Predvsem se uporabljajo na področju slik, kjer je zelo pomembno prepoznavanje objektov in obrazov, razvrščanje slik v skupine glede na podobnost, prepoznavanje gest in barvanje slik [15].

Nevronska mreža je v osnovi funkcija $f(x)$, ki preslika vhod x v izhod y .

Med postopkom učenja je ta funkcija optimizirana tako, da najde najboljšo aproksimacijo realnih podatkov [15]. Nevronska mreža je struktura, ki je sestavljena iz več nivojev. Nivoje si lahko predstavljammo kot vrsto vozlišč, ki se odzovejo v primeru, da je vzburenje na njih zadovoljivo - odvisno od aktivacijske funkcije. Struktura vozlišča in nivojev je predstavljena na sliki 2.1. Vozlišče pomnoži vsak vhod s trenutnimi vrednostmi uteži, doda še bias, vrednosti sešteje in moč aktivacije izračuna s pomočjo tako imenovane aktivacijske funkcije, ki tvori izhod vozlišča. Aktivacijski funkciji rečemo tudi neliearnot, saj poskrbi za to, da nevronska mreža ni le linearна funkcija [16]. Uteži se skozi postopek učenja spreminjajo in s tem določijo aktivacijo vozlišča.



Slika 2.1: Leva slika prikazuje zgradbo enega vozlišča, ki ima zasnovno podobno nevronom v možganih. Vhod je lahko izhod prejšnjega nivoja ali vhodni podatki v mrežo, ki se potem pomnožijo z utežmi in seštejejo. Aktivacija poskrbi, da se vozlišče odzove, ko je vzburenje dovolj veliko. Desna slika prikazuje zgradbo več nivojske nevronske mreže. V našem primeru ima ta en vhodni nivo, en skriti nivo in izhodni nivo. Iz: Introduction to Deep Neural Networks url<https://deeplearning4j.org/neuralnet-overview> in Neural Networks, urlhttp://docs.opencv.org/2.4/modules/ml/doc/neural_networks.html (dostopano: 21. junij 2017).

Nivojem v nevronskih mrežah, ki se nahajajo med vhodnim in izhodnim nivojem rečemo *skriti nivoji* (angl. *hidden layers*) [16]. Tradicionalni algoritmi na področju strojnega učenja so sestavljeni iz vhodnega, izhodnega

nivoja in enega skritega nivoja, *globoka nevronska mreža* (ang. *deep neural network*) pa ima vsaj dva skrita nivoja [17], večinoma pa mnogo več. Vsak nivo globoke nevronske mreže prepozna določen lastnosti vhodnih podatkov. Nivoji, ki se nahajajo globje lahko prepozna bolj kompleksne lastnosti podatkov, saj na vhodu, dobitjo lastnosti oziroma aktivacije nivoja pred njim.

Da nevronska mreža daje zadovoljive rezultate je potrebno utežem določiti prave vrednosti. To naredimo s postopkom učenja. Vsaka nevronska mreža ima cenilno funkcijo (ang. *loss function*), ki pove kako dobre rezultate na testnih podatkih nevronska mreža daje trenutno. V postopku učenja je cilj zmanjšati vrednost cenilne funkcije z enim od algoritmov optimizacije.

2.2.1 Konvolucijske nevronske mreže

Ker bi bilo na primeru slik pri uporabi klasičnih nevronskih mrež hitro preveč parametrov, kar bi poleg podaljšanja časa učenja povzročilo tudi prekomerno prilagajanje (ang. *overfitting*) in pomanjkanje pomnilnika, uporabljamo za take primere konvolucijske nevronske mreže. Te so zelo podobne običajnim nevronskim mrežam. Sestavljene so iz nevronov, ki imajo svoje uteži in bias, ki so učljivi. Operacije znotraj nevrona so podobne tistim pri običajnih nevronskih mrežah, le da so prilagojene pričakovanim vhodnim podatkom - slikam. Vhod v vsak nivo nevronske mreže je torej tenzor z obliko $\text{širina} \times \text{višina} \times \text{globina}$, globina pomeni število barvnih kanalov slike [18]. Konvolucijske nevronske mreže so v osnovi sestavljene iz treh vrst nivojev:

- **Konvolucijski nivo** je glavni gradnik konvolucijske nevronske mreže. Parametri tega nivoja so sestavljeni iz majhnih konvolucijskih jeder, ki pokrivajo majhno polje v širino in višino. Več takih jeder pa pokrivajo celotni nivo v globino, tako si lahko eno jedro predstavljamo kot utež pri skritih nivojih v običajni nevronski mreži, ki zravn izvaja še konvolucijo. Med prehodom po nevronski mreži izvedemo konvolucijo po celotni višini in širi vhodnega tenzorja, po globini pa se te izhode teh konvolucij sešteje enako kot pri običajni nevronski mreži. Izhod konvolucije z enim setom jeder je dvodimensionalna matrika. [19]

- **Pooling nivo** je namenjen pod-vzorčenju (ang. downsampling) na določenem nivoju. S tem zmanjšamo število parametrov, kar vpliva zmanjšanje računske zahtevnosti in prekomernega prilagajanja. Deluje na principu, da je točna lokacija značilke manj pomembna kot približna lokacija glede na ostale značilke. [20]
- **Polno povezni nivo** je nivo enak skritim nivojem pri klasični nevronski mreži. Večinoma se uporabi za zadnjih nekaj nivojev pri konvolucijski nevronski mreži, če je to primerno za dano nalogu.

2.3 Predstavitev slikovnih podatkov in barvni prostori

Slike, ki jih uporabljamo za učenje so shranjene v *RGB* [21] barvnem prostoru. Kot je pokazano v [14] se izkaže, da barvanje na osnovi prostora RGB daje slabše rezultate, saj prostor ni direktno primeren za učenje algoritmov za barvanje iz dveh razlogov:

- **Sistem se ne ujema dobro s človeško percepциjo barv**, saj so razdalje med enako sorodnimi barvami različne glede na odtenek [22]. Na primer, če imamo dva para barv: rdečo in svetlo rdečo, ter modro in svetlo modro, pri čemer sta barvi v obeh parih za človekov vizualni sistem enako različni, sta razdalji v RGB barvnem prostoru različni .
- **Nima ločenega kanala za svetlost** [21]. Glede na to, da modeli za barvanje napovedujejo le barvne elemente v sliki, svetlost pa se vzame iz originalne slike, je najbolj priročno, če uporabljamo barvni prostor, ki ima ločen kanal za svetlost, saj je izhod metode kar združena komponenta za svetlost z barvnimi komponentami.

2.3.1 Izbira primernega barvnega prostora

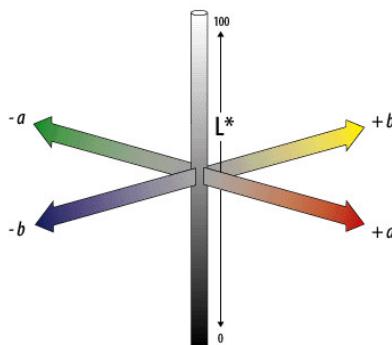
Na podlagi teh predpostavk je izbira prostorov omejena na *Lab* [23], *YUV* [24] in *HSL* [21]. Vsi ustrezajo drugi predpostavki iz 2.3. Edini, ki zares ustreza prvi predpostavki je *Lab*. Iz ugotovitev iz sorodnih del [14, 11, 13] se tudi najbolje izkaže prostor *CIE L*a*b**.

Obstaja več implementacij barvnega prostora *Lab*, ki vse težijo k dobri aproksimaciji človeškega zaznavnega sistema. Trenutno se najbolj uporablja *CIE L*a*b**, ki naj bi bil najboljša aproksimacija človeškega vizualnega sistema [22]. Prostor ima tudi to prednost, da je neodvisen od naprave. Prostor *CIE L*a*b** predstavi vse barve, ki jih je možno zaznati z tremi barvnimi kanali. *L** predstavlja svetlost, *a** se razširja od zelene proti rdeči barvi in *b** od modre proti rumeni. Prostor je grafično prikazan na sliki 2.2 *L** ser razteza od 0, ki predstavlja črno barvo, do 100, ki predstavlja belo barvo [25]. *a** in *b** komponenti nimata uradne omejitve, vendar sta v implementacijah ponavadi omejene na vrednosti v intervalu $[-128, 127]$, kar je možno predstaviti z 8 bitnim celim številom [26]. Ker zaradi pretvorb iz barvnega prostora *RGB* vrednosti višje od 100 ali nižje -100 redko dosežemo smo opazili, da nekatere implementacije omejijo barvne komponente na interval $[-100, 100]$. Za pomoč pri implementaciji nevronske mreže smo sami preizkusili kakšen je dejanski interval barv pretvorjenih iz *RGB* barvnega prostora. Intervale si lahko pogledate v tabeli 2.1.

2.3.2 Pretvarjanje med *RGB* in *CIE L*a*b** barvnim prostorom

Za pretvorbo med prostoroma ni enostavne enačbe, saj je *RGB* barvni prostor odvisen od naprav, *CIE L*a*b** pa je neodvisen. Tako se pretvorba zgodi v treh korakih [27]:

1. **Pretvorba iz *RGB* v *sRGB* ali *Adobe RGB*,** saj sta ta barvna prostora neodvisna od naprave. Ta pretvorba je odvisna od naprave.



Slika 2.2: Slika prikazuje kanale barvnega prostora CIE $L^*a^*b^*$. L^* predstavlja svetlost, a^* se razteza od zelene barve v najbolj negativni točki proti rdeči barvi, b^* ser razteza od modre proti rumeni. Nasprotajoče barve na kanalih a^* in b^* se nikoli ne kombinirajo v odtenek. Iz: Adobe, Technical Guid, CIELAB, http://dba.med.sc.edu/price/irf/Adobe_tg/models/cielab.html (dostopano: 24. junij 2017)

Slike, ki jih bomo uporabili v našem delu so že v *sRGB* obliku, saj so bile pretvorjene, ko so bile zajete z napravo.

2. **Pretvorba v CIE 1931 barvni prostor** ali drugače imenovan *CIE XYZ* barni prostor. Ta pretvorba se izvede s pomočjo linearne pretvorbe z matriko. Matrika je odvisna od izibire referenčne bele barve. Običajno se izbere referenčno temperaturo belo točke *D65*, ki je tudi standardizirana¹[28].
3. **Pretvorba iz CIE XYZ v $L^*a^*b^*$** se izvede z uporabo transformacijskih enačb opisanih v [29].

¹Zapis na uradni strani komisije International Commision on Illumination (krajše CIE), ki je postavila standard pravi, da se kot standardno uporablja referenčno temperaturo bele točke D65: http://cie.co.at/index.php?i_ca_id=484

Tabela 2.1: Največje in najmanjše vrednosti posamezne komponente CIE $L^*a^*b^*$ barvnega prostora, pri pretvorbi barv iz barvnega prostora RGB. Pretvorba je bila narejena z uporabo osvetlitve $D65$, ki določa temperaturo bele točke. Izkaže se, da je večina vrednosti komponent a^*b^* znotraj intervala $[-100, 100]$.

Kanal	Najmanjša vrednost	Največja vrednost
L^*	0	100
a^*	-86.185	98,254
b^*	-107.863	94.482

Poglavlje 3

Barvanje črno-belih slik z globokimi nevronskimi mrežami

V tem poglavju predstavljamo arhitekture nevronskih mrež, ki smo jih načrtovali, pogledali si bom pristope z regresijo in klasifikacijo in predstavili postopek učenja. Opisan je eksperiment z barvanjem večjih slik od tistih na katerih je bila mreža naučena, predstavljeni so učni in testni podatki ter način evalvacije.

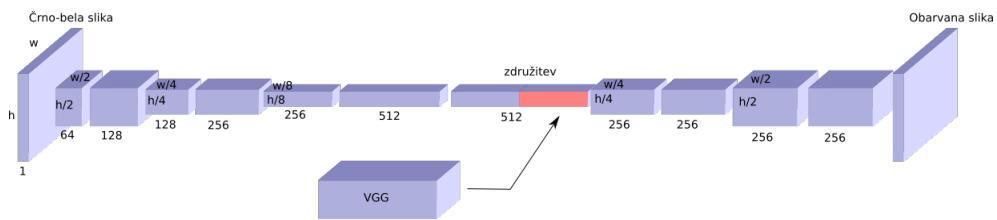
3.1 Arhitekture

V okviru magistrske naloge smo implementirali štiri arhitekture nevronskih mrež, kasneje smo te arhitekture kombinirali z različnimi cenilnimi funkcijami, pristopi (regresijski ali klasifikacijski) in načini napovedovanja (napovedovanje po delih ali na celi sliki).

3.1.1 Plitva arhitektura z globalno mrežo

Plitva arhitektura z globalno mrežo je sestavljena iz dveh delov, ki se kasneje združita v enotno mrežo. Glavni del predstavlja zaporedje konvolucijskih nivojev, ki na vhodu vzamejo sivinsko sliko z enim kanalom, izhod pa je obarvana slika z dvema kanala za barvni komponenti a^* in b^* . Po prvih

osmih kovolucijskih nivojih se mreža združi s tako imenovano *globalno mrežo*, ki prepozna objekte na sliki. Za globalno mrežo smo vzeli že naučeno šestnajst-nivojsko mrežo VGG-16 [10], ki smo ji odvzeli zadnji polno-povezani nivo in ji dodali nov polno-povezani nivo z izhodnim tenzorjem dolžine 256. Ker je arhitektura VGG-16 namenjena sprejemu barvnih RGB slik smo vhod prilagodili tako, da sprejme sivinsko sliko na vsakem od treh vhodnih kanalov. Arhitektura nevronske mreže je predstavljena na sliki 3.1 in v tabeli B.1 v prilogi.

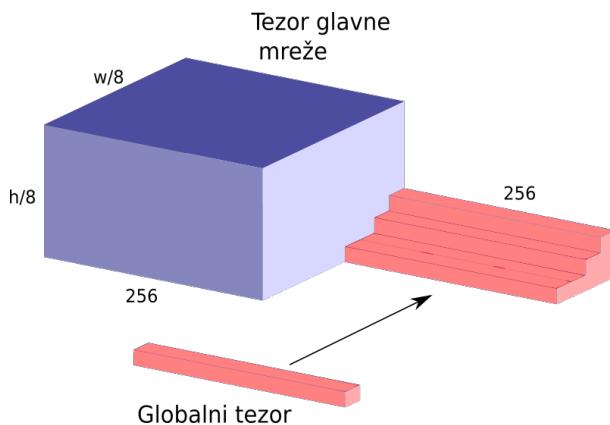


Slika 3.1: Slika prikazuje velikosti tenzorjev skozi plitvo arhitekturo z globalno nevronske mrežo. w in h predstavljata širino in višino vhodne slike. Podrobnosti nivoja združitev so predstavljene na sliki 3.2, nivo izhodne slike ni natančneje označen, saj se razlikuje v različnih implementacijah, ki so podrobnejše opisane v poglavjih 3.2 in 3.3.

Na tem mestu bi bilo smiselno opisati še združevanje glavne in globalne mreže. Vhod v element za združevanje sta tenzorja velikosti $\frac{w}{8} \times \frac{h}{8} \times 256$ iz glavne mreže in enodimenzionalni tenzor velikosti 256 iz globalne mreže. Pri tem w in h predstavljata širino in višino vhodne slike v mrežo. Pri združevanju vsakemu elementu širine in višine prvega tenzorja pridružimo tenzor globalne mreže, kot prikazuje slika 3.2. Tako na izhodu dobimo tenzor velikosti $\frac{w}{8} \times \frac{h}{8} \times 512$.

3.1.2 Globja arhitektura z globalno mrežo

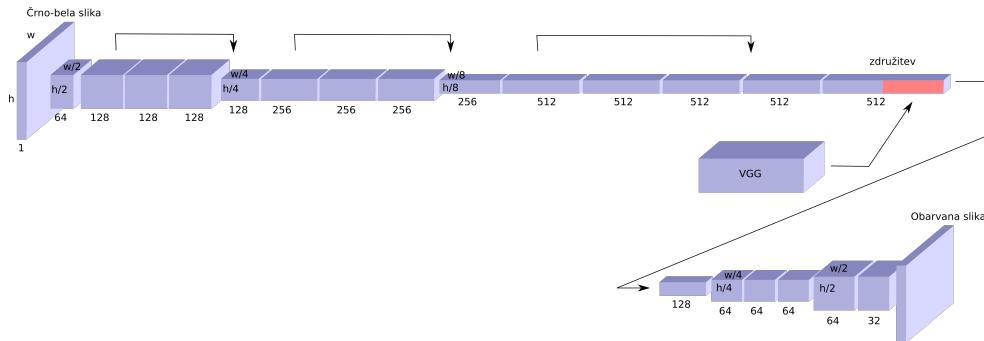
Ta arhitektura ima v osnovi enako zasnovo, kot arhitektura v poglavju 3.1.1. Razlika se pojavi pri globini glavne mreže. Ta ima namreč 14 konvolucij-



Slika 3.2: Prikaz delovanje nivoja združevanja glavne nevronske mreže z globalno nevronske mrežo. K izhodu glavne nevronske mreže (modri tenzor) dodamo tenzor iz globalne mreže (rdeči tenzor), tako da ga priključimo k vsem prostorskim lokacijam glavnega tenzorja kot dodatnih 256 kanalov. Izhodni tenzor ima tako 512 kanalov.

skih nivojev pred združitvijo in 7 po združitvi kot lahko vidite na sliki 3.3 in tabeli B.2 v prilogi. Za razliko od arhitekture opisane v poglavju 3.1.1 ta za zmanjševanje prostorskih (ang. *spatial*) dimenzijs uporablja *maksimalno združevanje* (ang. *max pooling*) [20] in za povečevanje le teh v zadnjih nivojih uporablja *transponirano konvolucijo* (ang. *transpose convolution*) [30] imenovano tudi dekonvlucija. Združevanja glavne in globalne mreže se izvede na način opisan v poglavju 3.1.1.

Ta arhitektura prinaša še eno spremembo. To so tako imenovane rezidualne povezave, ki so bile prvič uporabljene v nevronske mreži ResNet zasnovani s strani Microsoft Research [31], ki je leta 2015 zmagala na tekmovanju ImageNet [12]. Te povezave so na sliki 3.3 označene s puščicami nad nevronske mreže in predstavljajo povezavo, ki na mestu kamor kaže puščica, združi trenutni tenzor z tenzorjem izračunanim pred dvema nivojema. Operacija združevanja je seštevanje isto ležečih elementov v tenzorju.



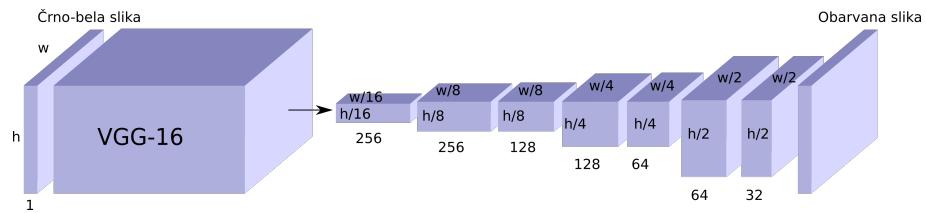
Slika 3.3: Slika prikazuje velikosti tenzorjev skozi globjo arhitekturo z globalno nevronsko mrežo. w in h predstavljata širino in višino vhodne slike. Podrobnosti nivoja združitev so predstavljeni na sliki 3.2, nivo izhodne slike ni natančneje označen, saj se razlikuje v različnih implementacijah, ki so podrobneje opisane v poglavjih 3.2 in 3.3.

3.1.3 Globja arhitektura brez globalne mreže

Ta arhitektura je enaka tisti opisani v poglavju in prikazani na sliki 3.3 v glavnem delu in se razlikuje po tem, da nima globalne mreže. Torej nima mreže, ki se pridruži v nivoju združitve, zaradi tega smo tudi ta nivo izpustili. Mreža za vhod vzame črno-belo sliko z enim kanalom in izračuna barvno sliko na izhodu. Ta arhitektura je bila načrtovana z namenom, da se preveri če ima globalna mreža prisotna v arhitekturah predstavljenih v poglavjih 3.1.2 in 3.1.1 kakšen vpliv na rezultate.

3.1.4 Dopolnjena VGG-16 arhitektura

Arhitektura je zgrajena tako, da za vrh mreže uporabimo mrežo VGG-16 [10], kateri smo odstranili vse polno povezane nivoje. Arhitektura je narejena tako, da na vhodu sprejme črno-belo sliko, ki jo potem prilagodimo za vhod mreže VGG-16, tako da ima tri vhodni kanale. Te dobimo tako, da vzamemo sivinsko sliko za vsak vhodni kanal. Tenzor, ki ga vrne zadnji konvolucijski nivo mreže VGG-16 podamo na vhod lastne mreže, ki je prikazana



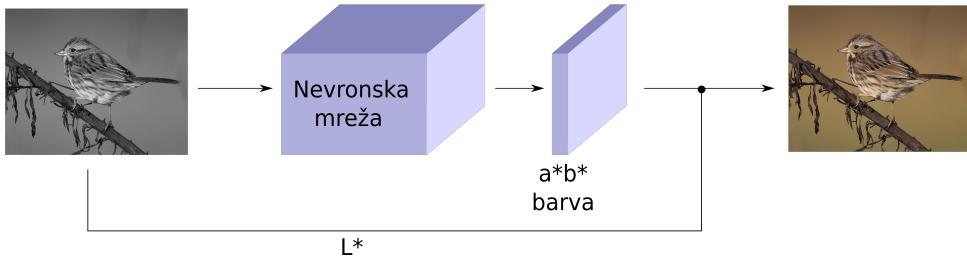
Slika 3.4: Slika prikazuje velikosti tenzorjev skozi dopolnjeno VGG-16 arhitekturo. w in h predstavlja širino in višino vhodne slike. Nivo izhodne slike ni natančneje označen, saj se razlikuje v različnih implementacijah, ki so podrobneje opisane v poglavjih 3.2 in 3.3. Prvi večji blok predstavlja mrežo VGG-16 [10]

na sliki 3.4 in podrobno opisana v tabeli B.3 v prilogi. Lastna mreža ima še 8 konvolucijskih nivojev in 4 nivoje nad-vzorčenja, ki poskrbijo za povečanje dimenzijske prostorske komponente. Po zadnjem nivoju izvedemo še povečanje slike za faktor 2, saj se nad-vzorčenje izkaže kot slabša možnost.

3.2 Pristopi z regresijo

V tem poglavju bomo prestavili regresijske metode, ki smo jih uporabili. Te se imenujejo tako, saj pri napovedovanju barve direktno napovejo vrednosti a^* in b^* barvne komponente v prostoru $CIE L^*a^*b^*$. Tu mreža predstavlja regresijsko funkcijo $y = f(x)$ za vsako točko slike, ki na vhod dobi točko sivinske slike x , izhod pa je kar vrednost y , ki predstavlja določeno barvo v našem primeru sta to dve vrednosti a^* in b^* .

Na sliki 3.5 je prikazan postopek delovanja, ki je skupen vsem regresijskim metodam opisanim v nadaljevanju. Da dobimo barvno sliko mreži podamo sivinsko sliko, mreža nato oceni barvni komponenti a^* in b^* , izhod mreže nato združimo z sivinsko sliko, ki je obenem komponenta L^* . Rezultat je obarvana slika.



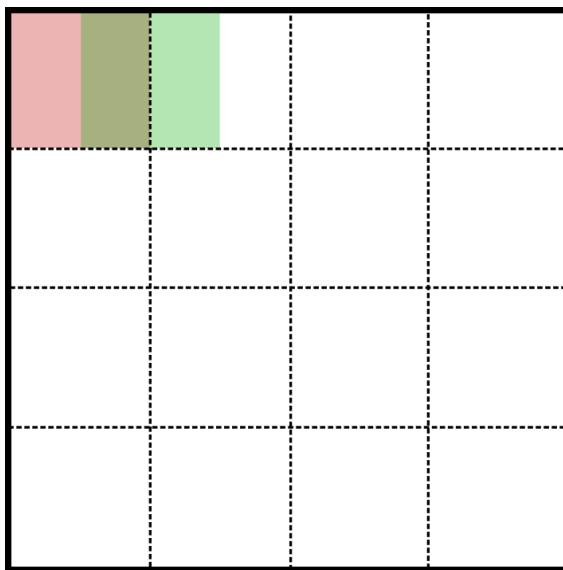
Slika 3.5: Shematski prikaz delovanja regresijske metode, ki na vhod prejme črno-belo sliko, s pomočjo nevronske mreže izračuna barvni komponenti a^* in b^* barvnega prostora $CIE L^*a^*b^*$ ter to združi s sivinsko sliko L^* , da dobi obarvano sliko.

3.2.1 Pristopi na delih slik

V to skupino lahko uvrstimo tri pristope, ki imajo skupno to, da smo barvanje izvajali na majhnih delih slik. Ta princip smo razvili z opazovanjem delovanja človeškega zaznavnega sistema, ki se bi lotil barvanja po delih, na sliki bi zaznal objekte in jih ločenoobarval. Na primer najprej vodo, nato gozd in kasneje še nebo.

Sliko smo preden smo jo podali nevronski mreži razdelili na koščke velikosti 32×32 slikovnih točk. Pri tem sta se sosednja koščka prekrivala za 16 slikovnih točk kot je prikazano na sliki 3.6. V arhitekturah, kjer je bila prisotna ločena globalna mreža, je ta še vedno na vhodu prejela celotno sliko, s katero smo pridobili globalni koncept slike. Dele slik smo kasneje sestavili z metodo prekrivanja, tako da so imele vrednosti točk pri robu manjši vpliv kot tiste pri sredini. Vpliv barvne točke se je izračunal po enačbi 3.1, kjer x predstavlja vrednost slikovne točke, d predstavlja oddaljenost od središča v številu slikovnih točk. Enačba se ločeno uporablja v vertikalni in horizontalni smeri. Da smo dobili končno vrednost v določeni točki smo sešteli vse prispevke za tisto točko utežene po enačbi 3.1.

$$y = \frac{d}{16}x \quad (3.1)$$



Slika 3.6: Shema prikazuje razdeljevanje slike velikosti 128×128 slikovnih točk, ki jo v vsaki smeri razdelimo na 7 enakih delov velikosti 32 slikovnih točk, ki se med seboj v vseh smereh prekrivajo za 16 slikovnih točk. Rdeč in zelen kvadrat prikazujeta prva dva dela s konceptom prekrivanja. Enak sistem uporabimo v vertikalni smeri.

Z takim načinom dela pričakujemo pohitritev učenja, saj menimo, da je že del slike dovolj, da se mreža nauči celotne tekture objekta. Na primer, da se naučimo barvati vodo ne potrebujemo celotnega območja vode na sliki, ki včasih lahko prekriva tudi pol ali več slike, ampak le en del. Z učenjem mreže na manjših vhodnih tenzorjih, kar dosežemo z učenjem po delih, se čas, ki ga mreža porabi za učenje ene serije (ang. *batcha*) podatkov zmanjša.

V tabeli 3.1 so predstavljene podrobnosti vsakega od pristopov barvanja po delih.

3.2.2 Pristopi na celih slikah

Za primerjavo točnosti metod na delih slik s tistimi na celih slikah. Smo dve metodi opisane v poglavju 3.2.1 pretvorili v metode za barvanje na celih

Tabela 3.1: Regresijski pristopi po delih, njihove arhitekture, ki so podrobneje opisane v poglavju 3.1 in cenilne funkcije uporabljene za učenje. Cenilne funkcije so podrobneje predstavljene v poglavju 3.4.

Ime metode	Arhitektura	Cenilna funkcija
Reg. po delih	Globja arh. z glob. mr.	MSE
- brez softmax	Globja arh. z glob. mr.	MSE
- brez globalne mreže	Globja arh. brez glob. mr.	MSE

slikah. Uporabili smo enako arhitekturo, ki smo jo prilagodili tako, da na vhodu sprejme celotno sivinsko sliko in vrne celotno obarvano sliko. Tej smo dodali še eno metodo, saj zaradi večkratnega pomanjšanja prostorskih dimenzij znotraj arhitekture ne more biti realizirana na manjših delih slik.

V tabeli 3.2 so podrobno predstavljene arhitekture in cenilne funkcije teh pristopov.

Tabela 3.2: Regresijski pristopi na celih slikah, njihove arhitekture, ki so podrobneje opisane v poglavju 3.1 in cenilne funkcije uporabljene za učenje. Cenilne funkcije so podrobneje predstavljene v poglavju 3.4.

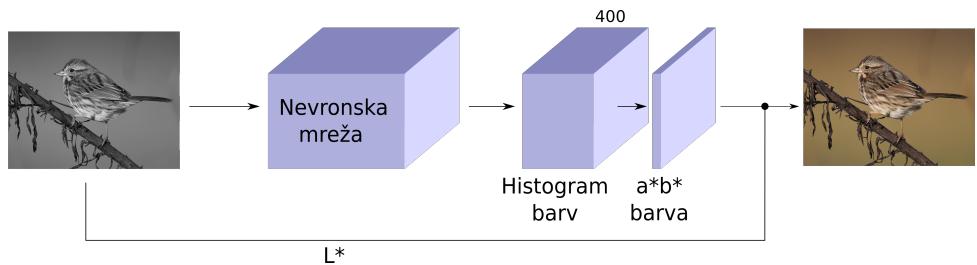
Ime metode	Arhitektura	Cenilna funkcija
Reg. cela slika	Globja arh. z glob. mr.	MSE
- brez globalne mreže	Globja arh. brez glob. mr.	MSE
Reg. cela slika VGG	Dop. VGG-16 arh.	MSE

3.3 Pristopi s klasifikacijo

Razvili smo štiri metode, ki namesto regresije uporabljata klasifikacijo. To so pristopi, kjer direktno ne napovemo številčne vrednosti barve ampak to

določimo s pomočjo klasifikacije v enega od razredov, ki predstavljajo nekaj sosednjih odtenkov v sliki. Klasifikacija se izvede z uporabo *softmax* funkcije v zadnjem nivoju mreže.

Kot je prikazano na sliki 3.7 je vhod v metodo črno-bela slika, ki se posreduje nevronske mreži, ki oceni obarvanje kot vektor verjetnosti za vsakega od razredov za vsako slikovno točko. Te vrednosti se potem pretvorijo v a^* in b^* komponento a^* in b^* . Enako kot pri regresiji je rezultat združitev sivinske slike z ocenjenimi barvnimi komponentami.



Slika 3.7

Razrede smo dobili tako, da komponenti a^* in b^* barvnega prostora *CIE L*a*b** razdelimo v 400 razredov. Vsako od komponent smo razdelili v 20 razredov med vrednostma -100 do 100 , kar pomeni da vsak razred zajema interval širine 10.

Za delovanje mreže moramo pretvoriti $a^* b^*$ zapis v histogram in obratno. Pretvorba iz zapisa $a^* b^*$ v histogram je potrebna saj moramo pri učenju referenčno sliko (ang. *ground truth*) pretvoriti v histogram, da lahko izračunamo napako. To izvedemo z enačbo 3.2, kjer a in b predstavljata a^* ali b^* vrednost slikovne točke, y pa indeks razreda v histogramu, ki zavzema vrednosti med 0 in 399 vključno.

$$y = 20 \left\lfloor \frac{a + 100}{10} \right\rfloor + \left\lfloor \frac{b + 100}{10} \right\rfloor \quad (3.2)$$

Pretvorba iz histograma v a^* in b^* se uporabi pri pretvorbi ocenjenih barvnih vrednosti s strani mreže in se izvede po enačbah 3.3. Pri tem a in b

predstavljata $a*$ in $b*$ barvne vrednosti slikovne točke, y predstavlja indeks razreda v histogramu, ki je bil napovedan z največjo verjetnostjo.

$$\begin{aligned} a &= 10 \left\lfloor \frac{y}{20} \right\rfloor - 100 + 5 \\ b &= 10(y \bmod 20) - 100 + 5 \end{aligned} \tag{3.3}$$

Pristope s klasifikacijo smo preizkusili, ker so v delih [13, 11] ugotovili, da so rezultati teh metod, slike, ki imajo močnejše in bolj realne barve. V tabeli 3.3 so prikazane metode s klasifikacijo, njihova arhitektura in cenilna funkcija.

Tabela 3.3: Klasifikacijski pristopi po delih, njihove arhitekture, ki so podrobneje opisane v poglavju 3.1 in cenilne funkcije uporabljene za učenje. Podrobnosti cenilnih funkcij možno najti v poglavju 3.4

Ime metode	Arhitektura	Cenilna funkcija
Klas. brez uteži - plitva arh.	Plitva arh. z glob. mr.	KL-divergenca
Klas. brez uteži - globja arh.	Globja arh. z glob. mr.	CE
Klas. z utežmi - plitva arh.	Plitva arh. z glob. mr.	CEw
Klas. z utežmi - globja arh.	Globja arh. z glob. mr.	CEw

3.4 Cenilne funkcije

Pri naših pristopih uporabljamo tri različne cenilne funkcije. Pri regresijskih pristopih se je izkazala za dobro cenilna funkcija *povprečna kvadratna napaka* (ang. *mean squared error*), ki jo označimo z oznako MSE in je predstavljena z enačbo 3.4. V enačbi Y predstavlja realno sliko, \hat{Y} oceno barv s strani mreže, h je višina slike, w je širina slike in c globina. MSE smo računali le

na komponentah $a*$ in $b*$, tako da $c = 2$.

$$MSE(\hat{Y}, Y) = \frac{1}{hwc} \sum_{h,w,c} (Y_{h,w,c} - \hat{Y}_{h,w,c})^2 \quad (3.4)$$

Pri enem od klasifikacijskih pristopov smo uporabili cenilno funkcijo Kullback-Leibler divergenca [32], ki jo označujemo z oznako KL -divergenca. V našem primeru smo izvedli Kullback-Leibler divergenco na posamezni slikovni točki in jo povprečili po celotni sliki kot prikazuje enačba 3.5. Oznake v enačbi so enake oznakam v enačbi 3.4, razen oznaka c v tem primeru predstavlja število kanalov, ki so v tem primeru razredi, v katere mreža klasificira barve, zato je $c = 400$.

$$KL\text{-divergenca}(\hat{Y}, Y) = \frac{1}{hw} \sum_{h,w} \sum_c Y_{h,w,c} \frac{Y_{h,w,c}}{\hat{Y}_{h,w,c}} \quad (3.5)$$

Zadnja cenilna funkcija je *križno entropija* (ang. *cross entropy*) [33] in jo bomo krajše označili z CE . Zaradi različnih implementacij uporabljam dve različici križne entropije. Prav je brez uteži in je predstavljana v enačbi 3.6, kjer so oznake enak oznakam iz zgornjih enačb. Križna entropija se prav tako kot Kullback-Leibler divergenca računa na razredih napovedanih s strani mreže.

$$CE(\hat{Y}, Y) = \sum_{h,w,c} Y_{h,w,c} \log \hat{Y}_{h,w,c} \quad (3.6)$$

Druga verzija križne entropije uporablja uteži, ki predstavljajo povprečno pogostost pojavitev posamezne barve na v slikah. Pogostost je bila izračunana na množici 100 tisoč naključnih slik iz zbirke ImageNet. Princip uteži v cenilni funkciji, da večji pomen tistim točkam, kjer se v originalni sliki pojavijo močnejše barve. S tem želimo prisilit mrežo, da bo večkratobarvala z močnejšimi barvami. Pri ostalih pristopih je namreč problem, da barve velikokrat niso tako močne in živahne kot bi lahko bile. Križno entropijo z utežmi označimo s CEw in je definirana z enačbo 3.7, kjer $w(Y_{h,w})$ predstavlja utež

za barvo v slikovni točki na koordinatah (h, w) .

$$CEw(\hat{Y}, Y) = \sum_{h,w} w(Y_{h,w}) \sum_c Y_{h,w,c} \log \hat{Y}_{h,w,c} \quad (3.7)$$

3.5 Postopek učenja

V tem poglavju bomo predstavili podrobnosti učenja na manjši množici, na večji učni množici in si za konec še pogledali kakšne značilke prepozna posamezni nivo nevronske mreže.

3.5.1 Učenje na manjši učni množici

Za učenje na manjši množici smo izbrali vse metode opisane v poglavjih 3.2 in 3.3. Za primerjavo natančnosti smo dodatno še implementirali metode razvite s strani R. Dahl-a [9], Iizuka in sod. [14] in Zhang in sod. [11].

Za posodabljanje parametrov smo uporabili Adam optimizator [34]. Pri tem so se za dobre izkazali naslednji parametri, ki so prikazani v tabeli 3.4. Pri vseh metodah smo uporabili velikost serije (ang. *batch size*) 32, razen pri treniranju metode Zhang in sod., kjer smo morali zaradi večjih dimenzij tenзорjev in posledično pomakanju pomnilnika uporabiti velikost serije 8.

Tabela 3.4: Parametri, s katerim smo nastavili Adam optimizator.

Parameter	Vrednost parametra
stopnja učenja (ang. <i>learning rate</i>)	10^{-4}
beta 1	0.9
beta 2	0.99
epsilon	10^{-8}

Pri treniranju smo beležili vrednosti cenilne funkcije tako na množici podatkov za treniranje in množici podatkov za validacijo po končanem vsakem

prehod čez vse podatke (ang. *epoch*). S tem smo opazovali kdaj je določena metoda optimalno naučena. To se v našem primeru zgodi v trenutku, ko vrednost cenilne funkcije na validacijski množici prenehajo padati ali celo začne naraščati. V tem trenutku je naša mreža optimalno naučena, zato smo te uteži uporabili za testiranje. Grafi padanja cenilnih funkcij za vse pristope so prikazani na sliki 3.8.

3.5.2 Učenje na večji učni množici

Za učenje na večji učni nožici smo izbrali sedem pristopov, pet lastnih in dva iz sorodnih del. Odločili smo se za tri regresijske pristope: *regresija po delih z globalno mrežo*, *regresija na celotni sliki z globalno mrežo* in *regresija na celotni sliki z mrežo VGG-16*. Izpustili smo oba pristopa brez globalne mreže, saj imata očitno slabše rezultate barvanja. Izbrali smo dva klasifikacijska pristopa *klasifikacija brez uteži - plitva arhitektura* in *klasifikacija z utežmi - plitva arhitektura*, namreč izkazalo se je, da globoka arhitektura le poslabša rezultate, zato smo se odločili, da izvedemo le primerjavo dveh plitvih arhitektur.

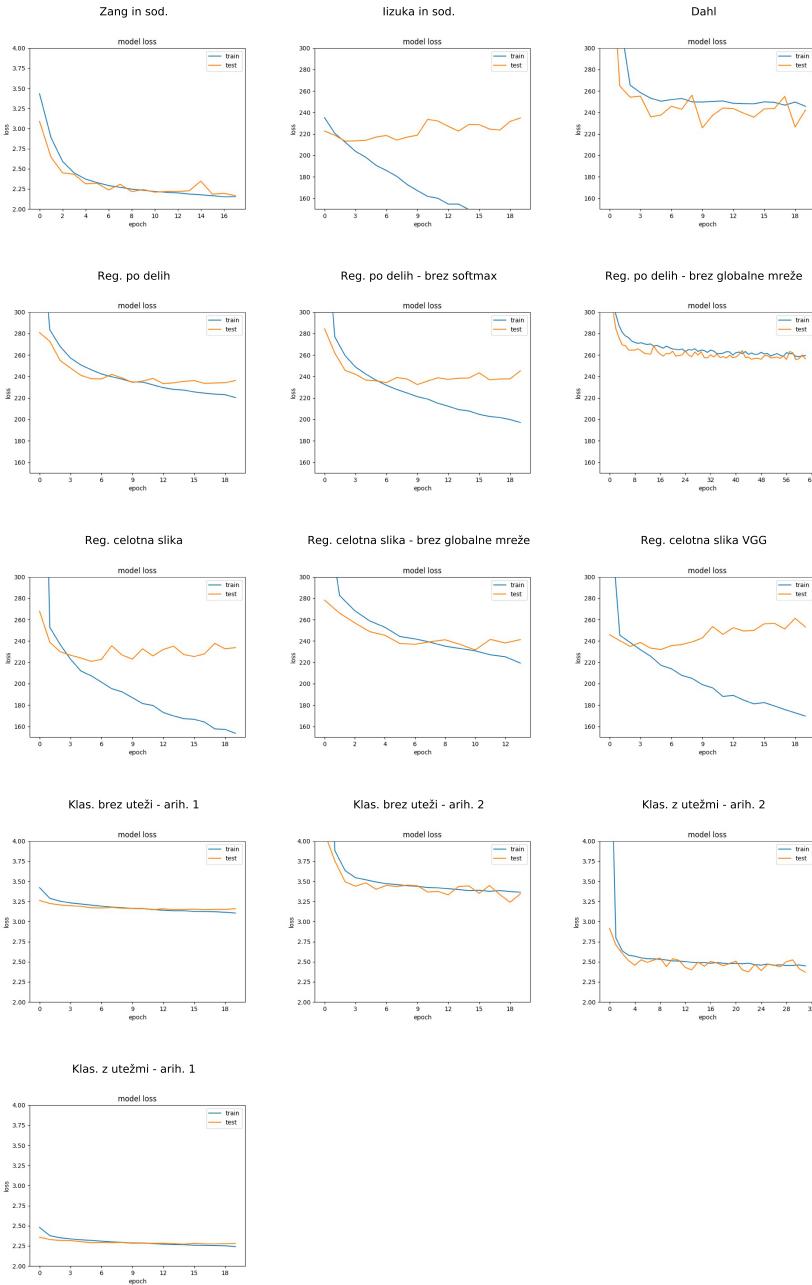
Za primerjavo tem pristopom smo dodali še pristopa Iizuka in sod. ter Dahl. Pristop, ki so ga predlagali Zhang in sod., se uči prepočasi, zato smo ga izpustili.

Za namen učenja smo uporabi enake parametre in optimizator, ki so opisani v poglavju 3.5.1. Ob učenju smo si enako, kot v primeru na manjši učni množici bolj izrisovali vrednost cenilne funkcije na množici za učenje in množici za validacijo, sprememjanje teh vrednosti je prikazano na sliki ??.

3.5.3 Pomen nivojev mreže

Konvolucijsko nevronsko mrežo si lahko predstavljamo, kot nivoje, ki poskrbijo za zajem značilk iz slike. V preteklosti so to počeli z različnimi pristopi kot so SIFT [35], HOG [35], SURF [36] in ostalimi. Nevronska mreža za to poskrbi, sama in izlušči tiste značilke, ki so za določeno nalogu relevantni.

POGLAVJE 3. BARVANJE ČRNO-BELIH SLIK Z GLOBOKIMI NEVRONSKIMI MREŽAMI



Slika 3.8: Prikaz padanja napake modela pri učenju. Za vsak prehod preko vseh podatkov (ang. *epoch*) je prikazana vrednost cenične funkcije na množici za treniranje in testni množici.

Uporabljene značilke lahko v grobem vidimo z vizualizacijo izhodov konvolucijskih nivojev.

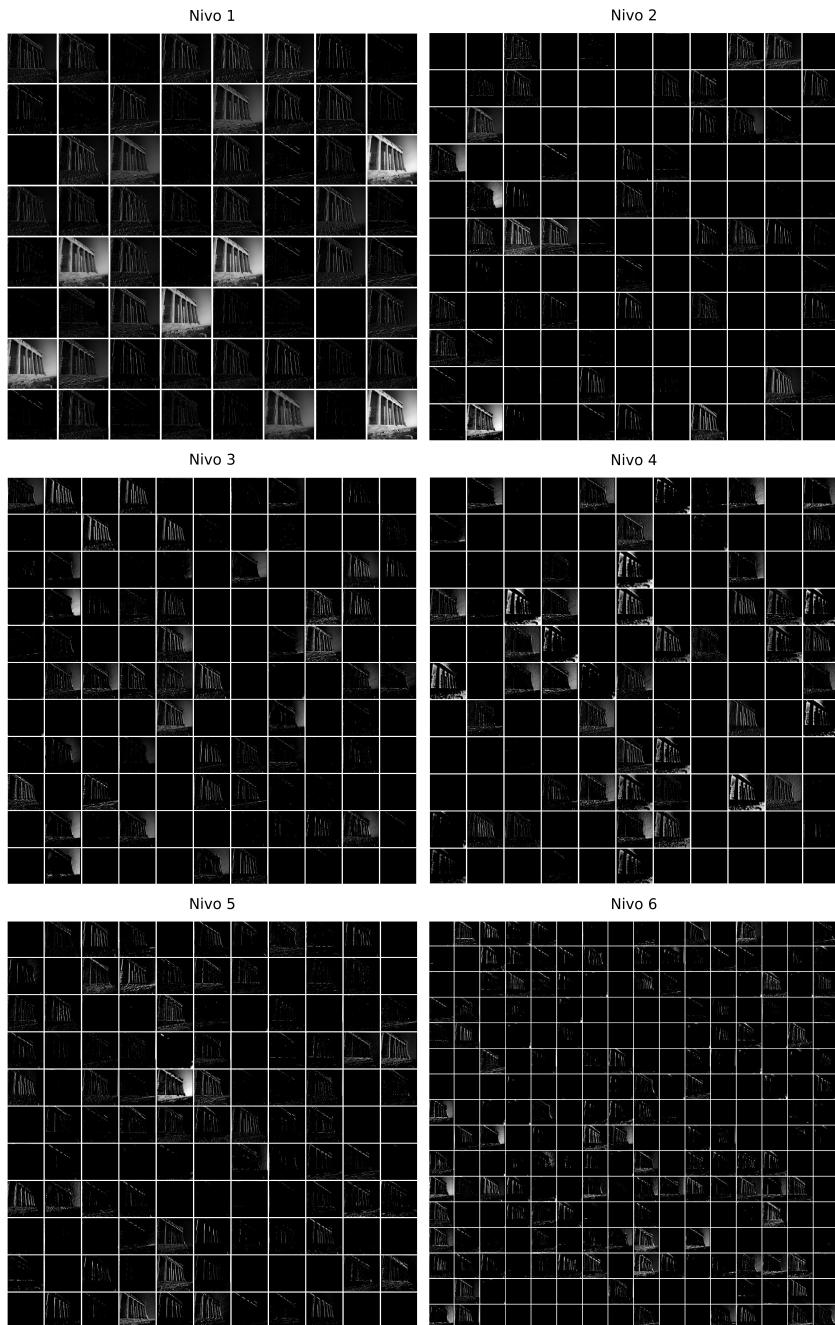
V tem poglavju bomo pogledali, katere značilke nivoji zaznajo v ta namen smo izrisali izhode prvih šestih konvolucijskih nivojev nevronske mreže, ki so prikazani na sliki 3.9. Vsak nivo predstavlja več slik, ki so izhodi posameznih filtrov, število slik je odvisno od števila filtrov. Čeprav ima mreža več nivojev smo se odločili, da prikažemo le prvih šest saj so ti najbolj razumljivi. Za vizualizacijo smo si izbrali pristop *regresija celotna slika z globalno mrežo*, saj je ta najbolj primerna za to vizualizacijo.

Slike povejo na kaj se filtri v posameznem nivoju odzivajo oziroma kaj zaznajo. V prvem nivoju lahko opazimo, da se v večji meri osredotočajo na robove v sliki. Opaziti je, da se nekateri odzovejo tudi na večji del slike in tako zgleda, kot da izhod kar osnovna slika. V drugem nivoju lahko opazimo že bolj osredotočene odzive. Nekateri še vedno zaznajo robove, drugi pa se že osredotočijo samo na določene dele slike na primer vodoravne robove ali navpične robove. Nekateri zaznajo tudi že dele, ki se enako obarvajo na primer nebo v ozadju.

V tretjem in četrtem nivoju je te osredotočenosti na določeno del ali motiv še več. Nekatere značilke je že težko razumeti in opisati. Še vedno je veliko filtrov, ki zaznavajo robove in površine, ki bodo kasneje enako obarvani. To se še vedno nadaljuje tudi v petem in šestem nivoju, kjer je še več značilk, ki imajo pomen za nevronske mreže, težje pa so razložljive nam ljudem. Opazimo lahko tudi, da se v kasnejših nivojih pojavlja tudi več izhodov z zelo malo ali praktično nič aktivacijami. Za te predvidevamo, da služijo drugačnim značilkam, ki jih v dani sliki ni bilo možno zaznati.

3.6 Barvanje večjih slik

Večina pristopov v sorodnih delih je naučenih za barvanje slik velikosti 224×224 in ima to omejitve, da omogoča le barvanje slik te velikosti. Iizuak in sod. [14] omogočajo barvanje večjih slik, tako da večjo sliko podamo enaki mreži



Slika 3.9: Vizualizacija izhodov prvih 6 konvolucijskih nivojev nevronske mreže na primeru slike Atenske akropole. Slika prikazuje kaj v sliki zaznajo posamezni nivoji in posamezni filtri.

na vhod, ki nima omejitev v velikost, saj je sestavljena le iz konvolucijskih nivojev. Pri tem je komentar avtorjev, da mreža deluje najbolje na slikah velikosti 224×224 .

Z pristopi po delih, ki smo jih implementirali v okviru tega dela, ta problem rešujemo drugače. Sliko katerekoli velikosti večje od 32×32 razdelimo na dele velikosti 32×32 s prekrivanjem, jo obarvamo po delih in potem spet sestavimo po principu opisanem v poglavju 3.2.1.

Primerjavo kakovosti barvanja večjih slik smo izvedeli tako, da smo pristopa *Iizuka in sod. ter regresijo po delih z globalno mrežo* preizkusili na istih slikah pomanjšanih na velikost 224×224 slikovnih točk, kjer naj bi bilo barvanje optimalno in velikost 896×896 . Primerjava je izvedena na mrežah naučenih na manjši učni množici s 100000 slikami.

3.7 Podatki

Podatke, ki smo jih uporabili v za treniranje in validacijo pristopov smo pridobili iz podatkovne zbirke ImageNet [12], ki vsebuje približno 14 milijonov slik. Iz zbirke smo naključno izbrali množico podatkov jih za namen učenja pretvorili v *CIE L*a*b** barvni prostor. Pri preizkušu na manjši množici smo za učenje naključno izbrali 100000 slik za vladacijo pa 10000 slik iz nabora. Validacijska množica je bila obenem tudi testna množica. Za učenje na večji množici smo naključno izbrali 2854912 slik, za validacijo smo še vedno uporabljali 10000 slik, ki so bile obenem tudi testna množica.

Za namen testiranja barvanja večjih slik opisanega v poglavju 3.6 podatki iz podatkovne zbirke *Imagenet* niso bili zadovoljivi, saj so slike večinoma velikosti manjših od 500×500 slikovnih točk. Odločili smo se, da testiranje izvedemo na lastnih slikah, ki so večje od velikosti 896×896 , kar pomeni, da slik ni potrebno povečevati in s tem povzročati dodatnih napak v barvanju zaradi slabe kvalitete slik. Za namen testiranja smo vzeli 584 slik, katere je aplikacija *Google Photos*¹ ocenila, da gre za slike pohodništva. Za te slike pohodništva smo se odločili, ker gre večinoma za slike narave, kjer je barvanje

ponavadi najboljše in lahko tako razliko opazujemo na slikah, ki se večinoma barvajo dobro.

3.8 Računanje napake

Za primerjavo metod smo napake računali na testni množici. Pri tem smo uporabili dve metrike: koren povprečne kvadratne napake (ang. *Root mean squared error*) in razmerje med signalom in šumom (ang. *Peak signal-to-noise ratio*).

Koren povprečne kvadratne napake (RMSE) za vsako sliko smo izračunali s pomočjo enačbe 3.8, kjer w in h prestavlja širino in višino slike ter c predstavlja število kanalov slike. d je originalna slika (ang. *ground truth*) in \hat{d} obarvana slika s strani pristopa za barvanje. Napaka je bila izračunana za vsako sliko posebej in kasneje povprečena preko vseh slik. Napaka RMSE je bila izračunana na slikah v CIE L*a*b* barvnem prostoru le za a^* in b^* barvni kanal, saj za L^* barvni kanal izračun napake ni smiseln, ker so vrednosti vzete iz originalne slike.

$$RMSE = \sqrt{\sum_{i=1}^h \sum_{j=1}^w \sum_{k=1}^c (d_{i,j,k} - \hat{d}_{i,j,k})^2} \quad (3.8)$$

Razmerje med signalom in šumom (PSNR) je metrika, ki kaže razmerje med največjo možno močjo signala in močjo šuma, ki signal pokvari. Primerna je za primerjave rekonstruiranih podatkov, kot so v našem primeru validacijske slike, ki jih poskušamo rekonstruirati s pomočjo pristopov, ki bazirajo na nevronske mrežah. Vrednosti razmerja med signalom in šumom se merijo v enoti decibel (dB) [37]. Zadovoljive vrednosti rekonstrukcije slike 8 bitnih podatkov, kar naši podatki so, saj smo primerjali slike v barvnem prostoru *RGB*, je med 30 in 50 dB [38].

Napako PSNR smo izračunali z enačbo 3.9, v kateri ima MAX_I največjo možno vrednost slikovne točke, kar je v barvnem prostoru RGB 255, $RMSE$

¹<http://photos.google.com>

pa je napaka izračuana z enačbo 3.8. Napako smo povprečili preko vseh slik v testni množici. Za izračun napake smo izbrali barvni prostor RGB, saj računanje PSNR v prostoru L*a*b* ni možno, ker ne poznamo največje možne vrednosti signala za komponenti a^* in b^* .

$$PSNR = 20 \log_{10} \left(\frac{MAX_I}{RMSE} \right) \quad (3.9)$$

*POGLAVJE 3. BARVANJE ČRNO-BELIH SLIK Z GLOBOKIMI
NEVRONSKIMI MREŽAMI*

Poglavlje 4

Rezultati in diskusija

V tem poglavju predstavljamo natančnosti pristopov naučenih na manjši množici in jih med seboj primerjali. Sledijo natančnosti na večji množici podatkov in primerjava pristopov pri barvanju večjih slik.

4.1 Primerjava metod na manjši učni množici

Tabela 4.1 prikazuje natančnost pristopov na testni množici slik. Izkaže se, da se na manjši množici najbolje obnese pristop Iizuka in sod., ki je glede na napako RMSE za 0.066 boljši od našega pristopa z regresijo na celih slikah. Ostali pristopi razviti s strani drugih avtorjev se na tej množici obnesejo slabše od večine naših pristopov.

Izkazalo se je tudi, da se na tej množici glede na napako regresijski pristopi obnesejo bolje kot pristopi s klasifikacijo. Pristopa regresija po delih se z klasifikacijo brez uteži z globoko, ki ima enako arhitekturo, glede na RMSE razlikuje skoraj za 2. Med pristopi z regresijo je opaziti boljše rezultate pri mrežah, ki barvajo celotno sliko na enkrat. Opazimo lahko tudi, da globalna mreža prinese izboljšave glede na RMSE od nekje 0.3 do 0.4. Pri klasifikacijskih pristopih se je izkazalo, da plitva arhitektura deluje bolje pri tej učni množici. Izkaže se tudi, da uteži v cenilni funkciji ne prinesejo izboljšave v natančnosti glede na napako.

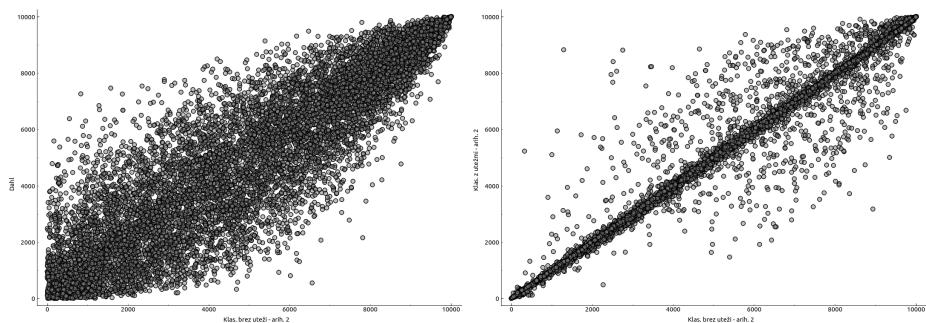
Tabela 4.1: Tabela prikazuje napake izračunane na testni množici podatkov. Za vsakega od pristopov smo izračunali dve napake opisane v poglavju 3.8. V zgornjem delu tabele so prikazane napake na pristopih iz sorodnih del, v vmesnem napake na pristopih z regresijo in v spodnjem delu tabele napake na pristopih s klasifikacijo.

Metoda	RMSE	PSNR
Zhang in sod.	15.004	22.252
Iizuka in sod.	12.941	23.439
Dahl	13.936	22.551
Reg. po delih	13.216	23.199
- brez softmax	13.206	23.183
- brez globalne mreže	13.767	22.840
Reg. celotna slika	13.007	23.434
- brez globalne mreže	13.334	23.068
Reg. celotna slika VGG	13.387	23.131
Klas. brez uteži - plitva arh.	14.336	22.738
Klas. brez uteži - globoka arh.	15.086	22.380
Klas. z utežmi - plitva arh.	14.573	22.610
Klas. z utežmi - globoka arh.	15.137	22.395

Za vsak pristop smo napake za slike iz testne zbirke spremenili v range, glede na napako RMSE na določeni sliki, ki se raztezajo od najboljše z rangom 0, do najslabše z rangom 9999. Izkaže se, da rangi med metodami močno korelirajo, kar pomeni da je natančnost barvanja slike v veliki meri odvisna od motiva na sliki.

Korelacijo med pristopi je možno videti na sliki 4.1. Na slik je prikazan graf, ki prikazuje odvisnost rangov med prvim pristopom na osi X in drugim pristopom na osi Y . Izkaže se, da je korelacijo v veliki meri prisotna pri vseh metodah, je pa različna glede na primerjane metode. Metodi na desni

sliki, ki sta si bolj podobni glede na arhitekturo in način barvanja imata zelo veliko korelacijo, ki je skoraj linearna funkcija. Metodi na levi slikici, ki sta si na način napovedovanja bolj različni, ena je regresijska in druga klasifikacijska imata manjšo korelacijo, ki pa je še vedno prisotna (še vedno so točke razporejene okoli premice, ki razpolavlja kvadrant grafa vendar je odstopanj več). Podobne slike dobimo tudi pri primerjavi ostalih pristopov.



Slika 4.1: Graf prikazuje rangiranje slik glede na napako $RMSE$ pri dveh različnih metodah. X os predstavlja rang pri prvi metodi, Y pa rang pri drugi metodi. Prva slika prikazuje primerjave rangov Dahlovega pristopa in klasifikacijskega pristopa z globjo arhitekturo. Druga slika prikazuje range pri dveh klasifikacijskih pristopih z enakimi arhitekturami.

Ker smo želeli podobnost metod med seboj primerjati v prostoru, smo izračunali Spearmanovo korelacijo rangov [39] za vsak par metode. Korelacije so številčno prikazane v tabeli A.1 v prilogi. Za izris podobnosti smo uporabili metodo večdimenzionalno skaliranje (ang. *Multidimensional scaling* - MDS) [40] na Spearmanivih korelacijsah.

Pristopi so glede na arhitekturo in vrsto pristopa razporejeni v prostor, ki je prikazan na sliki 4.2. Izkaže se, da se pristopi z klasifikacijo pojavijo desno v prostoru in so bolj oddaljeni od tistih z regresijo levo. Pri pristopih z klasifikacijo opazimo, da na različnost bolj vpliva vrsta arhitekture, kot uporaba uteži glede na pogostost barve. Izkaže se, da je pristop Zhang in sod. bližje tistim z globjo arhitekturo. Glede na to, da so lastnosti teh arhitektur popolnoma drugačne, se izkaže, da na podobnost glede na napake

vpliva predvsem globina arhitekture.

Pri regresijskih pristopih lahko opazimo, da je največja razlika glede na uporabo globalne mreže. Pristopi, ki ne uporabljajo globalne mreže so na vrhu prikaza ostali pa spodaj. Regresija celotna slika VGG je nekje na sredini med obema gručama. To gre verjetno pripisati dejству, da ta pristop uporablja mrežo VGG-16, ki je del globalne mreže pri ostalih pristopih, vendar ta pristop to mrežo izkorišča kot glavno.

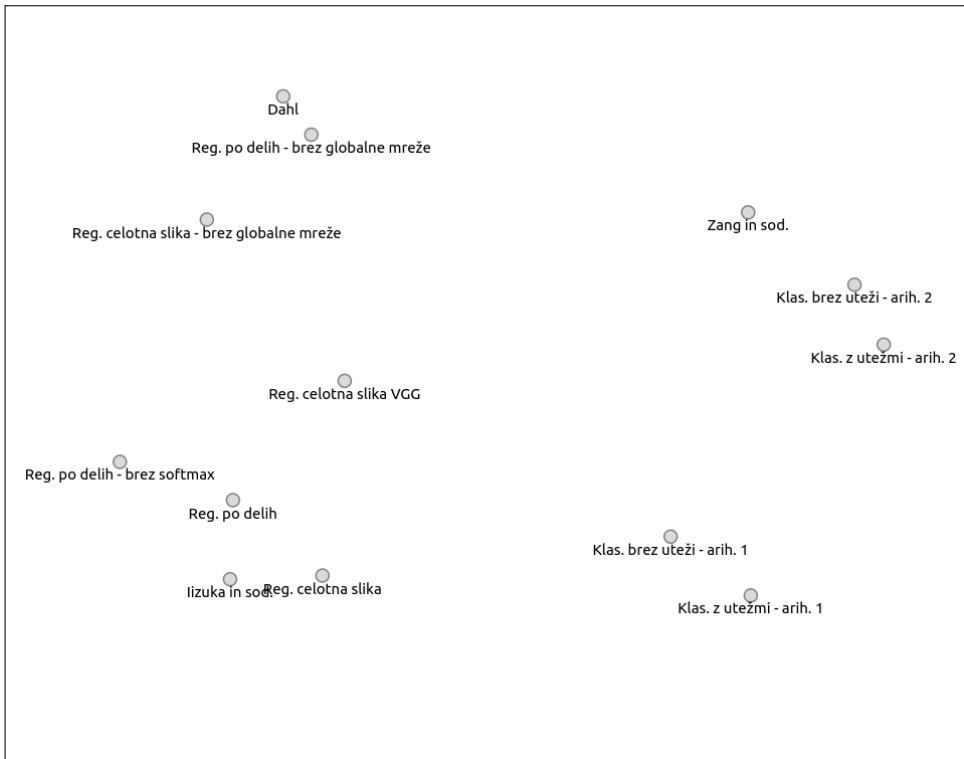
Čeprav je Dahlov pristop glede na arhitekturo popolnoma različen našim pristopom se glede na napake izkaže podoben pristopom brez globalne mreže. Pristop Iizuka in sod. je v gruči s pristopi, ki uporabljajo globalno mrežo, saj tudi sam uporablja podoben pristop. Pri pristopih z regresijo lahko opazimo še, da sta pristopa Iizuka in sod. ter pristop z regresijo na celi sliki bolj skupaj, saj oba delujeta na celotni sliki.

Zanimalo nas je tudi. Katere so tiste slike, kjer eden od pristopov opravi dobro delo, ostali pa mnogo slabše in obratno. Ob pogledu na sliko 4.1 lahko opazimo, da so taki primeri točke, ki ležijo najbolj stran od linearne premice, zato smo te slike našli z metodo za iskanje osamelcev (ang. *outliers*) [41].

Slike, ki najbolj izstopajo glede na napako na različnih pristopih, so prikazane kot točke v prostoru, ki ga dobimo z metodo MDS glede na napako, na sliki 4.3. Ob pregledu slik se izkaže, da gre tukaj večinoma za slike, kjer je težje zaznati teksturo ob tem predvidevamo, da so se določene metode bolje naučile ravno te teksture kot druge.

Ob bolj natančnem pregledu prostora smo ugotovili, da je napaka močno povezana s motivom na sliki. V prostoru so bližje skupaj slike s podobnim motivom in barvami. Na levi strani so prikazane tri slike, ki so blizu skupaj in jim je skupno to, da je na sliki morje ali nebo, ki sta oba modre barve in določen objekt (v našem primeru žival). Na desni strani sta dve slike, ki se tudi ujemata glede na odtenke v sliki, čeprav je motiv popolnoma drugačen.

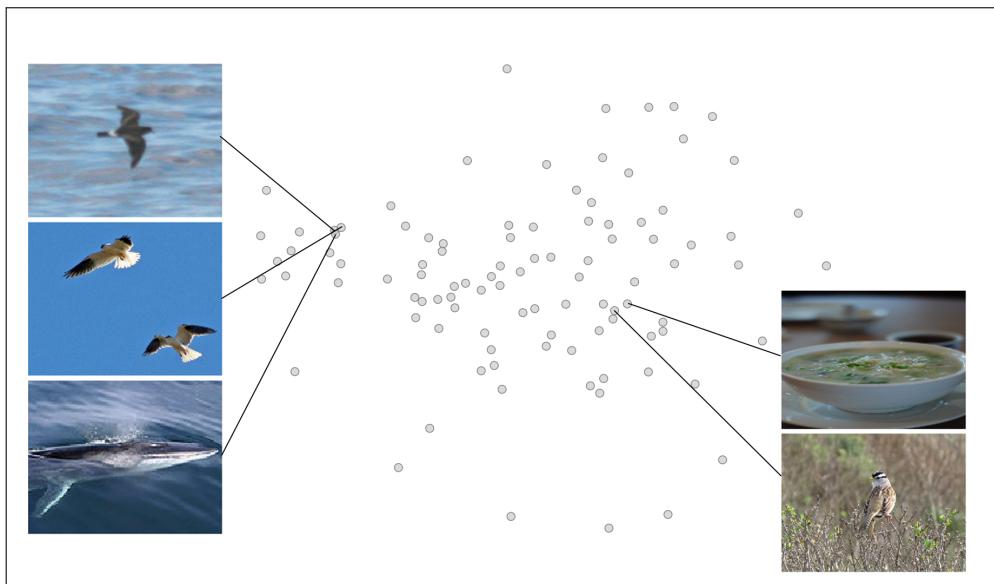
Ker pa se ukvarjam s slikami pa dodajamo še primerjavo barvanja na slikah. Primerjave po pristopih so prikazane na sliki 4.4 za regresijo in na sliki 4.5 za klasifikacijo. Slike smo izbrali, tako da prva dva stolpca prikazujeta dve



Slika 4.2: Primerjava metod v prostoru MDS kaže sorodnosti med metodami glede na vrsto pristopa (klasifikacija proti regresiji), arhitekturo mreže in načinom napovedovanja (na delih ali na celih slikah). Izkaže se največja razlika med vrsto pristopa, prav tako je velika razlika med arhitekturami in manjša je razlika med načinom napovedovanja.

slike iz množice 20 najbolje obarvanih s strani vseh algoritmov, 3 in 4 stolpec prikazujeta slike, ki so se bile z različno dobro obarvane s strani različnih pristopov. Te slike sta izbrane izmed točk v prostoru na sliki 4.3. Zadnja dva stolpca prikazujeta slike, ki so bile v množici 20 najslabše obarvanih s strani vseh algoritmov.

Opazimo lahko, da sta slike v prvih dveh stolpcih v skupini najbolje obarvanih zato, ker so že originalne slike z zelo malo barve, te so namreč zelo blede. Algoritmi, posebej regresijskih, ki običajno obarvajo z bolj nenasičenimi barvami so se zato zelo približali originalni slikam, čeprav barvanje v



Slika 4.3: Razporeditev slik v prostoru MDS, ki zajema 100 slik, pri katerih so natančnosti najbolj različne glede na rang pri različnih metodah. Prostor je bil izrisan glede na metodo MDS. Opazimo lahko, da so v prostoru so podobnih slik bliže skupaj. Dve taki podobni skupini slik sta prikazani ob robu.

več primerih ni ravno najboljše. Pri drugem in tretjem stolpcu so se nekateri pristopi dobro približali pravi barvi drugi se opazi, da barvanje s strani vseh algoritmov ni bilo enako dobro. Slike iz zadnjih dveh stolpcev sta bila glede na napako v množici najslabše obarvanih slik zato, ker imajo originalne slike zelo močne odtenke, katerim se pristopi niso približali, čeprav so nekatera barvanja dovolj naravna, če ne primerjamo z originalno sliko.

V primerjavi metod iz sorodnih del, pričakovano opazimo, da najboljše barva pristop Iizuka in sod., ki je imel tudi najmanjšo napako. Zang in sod. se na določenih delih obnese dobro, vendar so slike zelo lisaste in nepopolno obarvane, medtem ko so pri metodi Dahl odtenki zelo rjavi, čeprav vmes lahko opazimo nekaj pravih barv.

Pri primerjavi regresijskih pristopov lahko opazimo, da je po pričakovanjih

najboljše barvanje s strani Regresije na celotnih slikih z globalno mrežo, čeprav Regresija na delih slik ne zaostaja dosti. Opazimo lahko tudi pomen in izboljšavo z uporabo globalne mreže. Enaki pristopi brez globalne mreže so obarvali bolj nenatančno, nendaravno, prisotnih je tudi več rjavih odtenkov.

Pri klasifikacijskih pristopih opazimo, da pristopi z plitvo arhitekturo dajo boljše rezultate, kot tisti z globoko, kjer barvanja skoraj da ni. Pri primerjavi z utežmi in brez lahko zaznamo, da pristop z utežmi barva z močnejšimi odtenki, kot tisti brez, kar je bilo za pričakovati, saj je namen uteži, zmanjšati izbor bolj šibkejših odtenkov, ki imajo a^* in b^* vrednost bližje nič. Pristopi so nagnjeni k izbiri teh, ker se bolj pogosto pojavljajo v slikah. Kljub barvanju z močnejši odtenki barv in s tem približevanju realni barvi, je na pogled barvanje brez uteži bolj naravno, saj je opaziti manj napak v barvanju (na slikih s ptico sta les in ptica pobaranva bolj realno in letalo nima okoli sivinskega pasu).

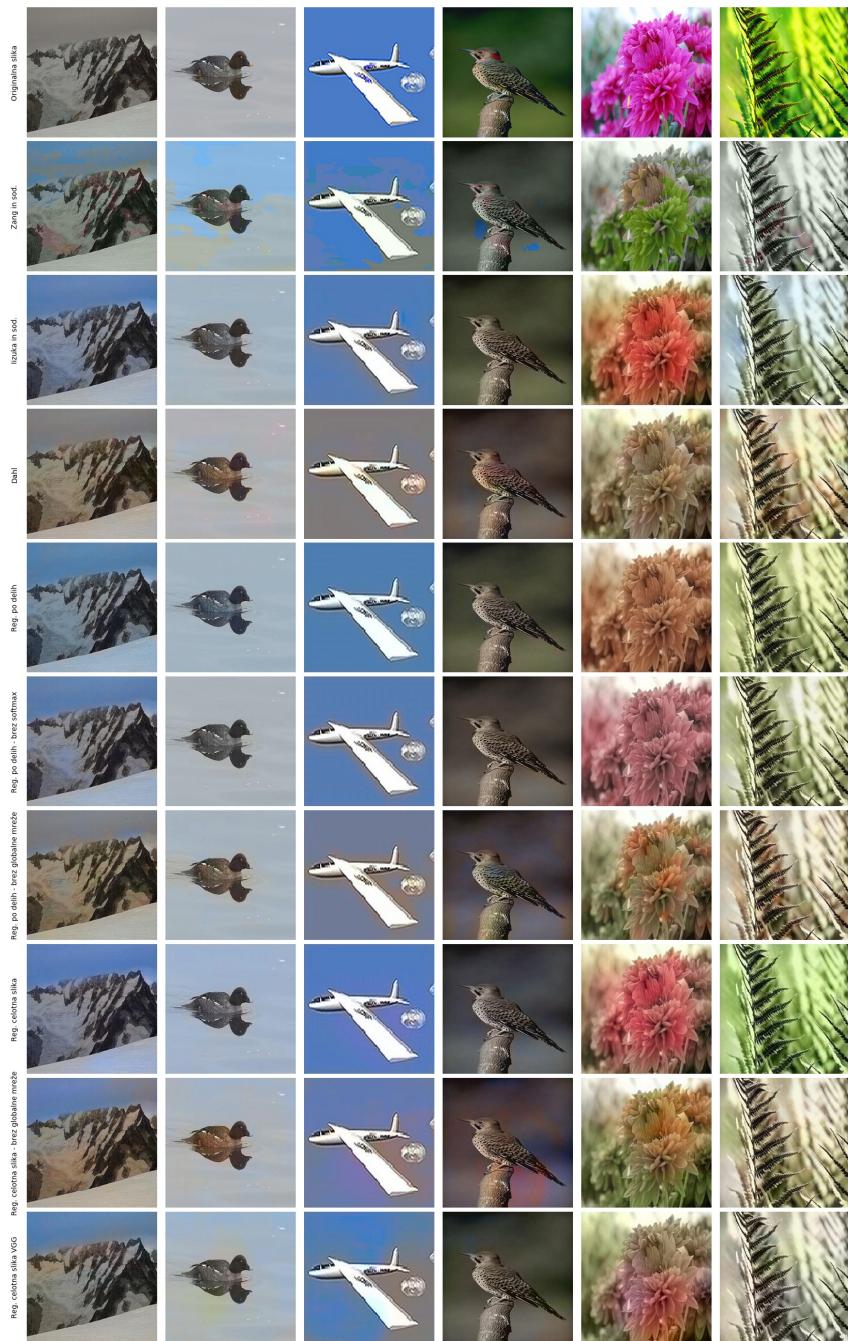
Pri primerjavi slik pristopi z regresijo obarvajo bolje in bolj naravno, kot pristopi z klasifikacijo, čeprav je nekaj izjem pri barvanju neba in praproti. Izkaže se tudi, da barvanje z regresijo večkrat obarva z bolj rjavimi odtenki, kar pri klasifikacijskih pristopih ni zaznati. Tam je bolj pogosto, da slika ni obarvana.

4.2 Primerjava metod na večji učni množici

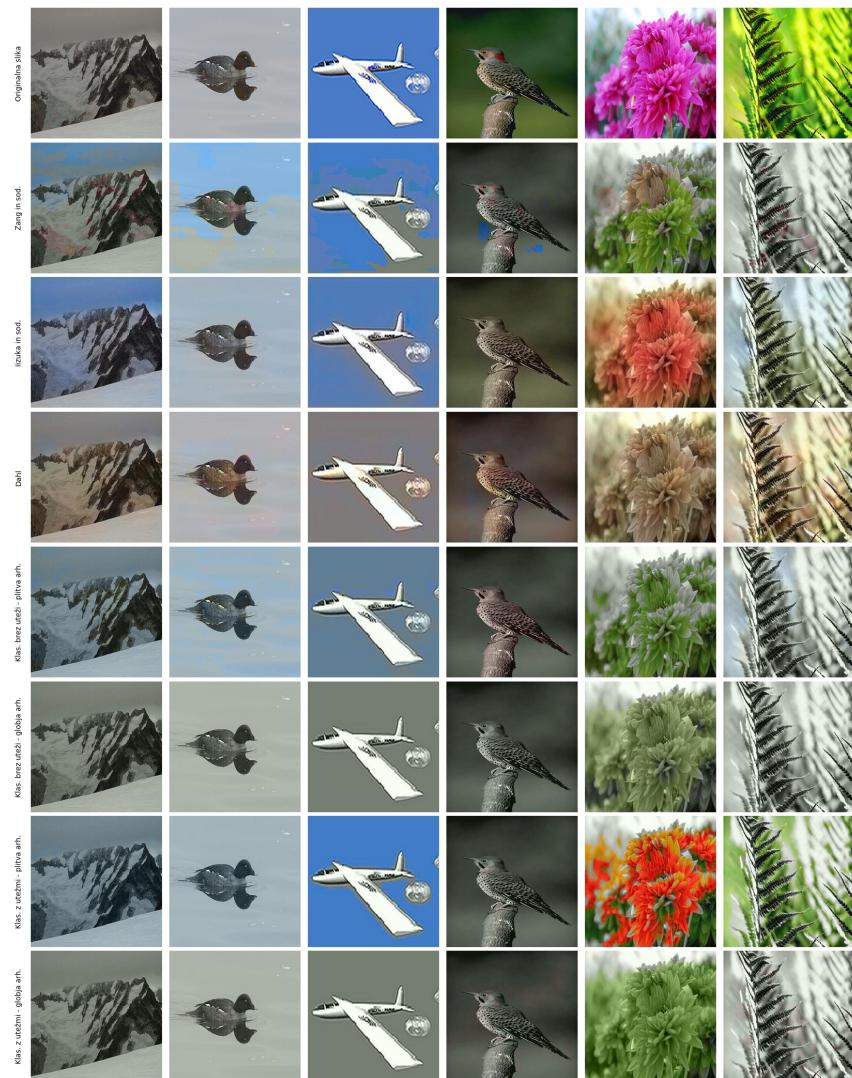
Tukaj bi primerjali metode, ki bi bile naučene na več slikah, tako bi pridobili tudi boljša barvanja za pokazat izboljševanje barvanja glede na epoche. Primerjavo bi izvedli na manj manj metodah.

4.3 Barvanje večjih slik

Tabela 4.2 prikazuje napake pri dveh velikostih slik na dveh metodah. Ena velikost je velikost na kateri je bila mreža naučena, druga pa je 4-kratna velikost slik v širino in višino. Izkaže se, da ima pristop *Regresija po delih z*



Slika 4.4: Slike iz testne množice, ki so bile obarvane z regresijskimi pristopi opisanimi v tem delu in pristopi iz sorodnih del. Vsaka vrstica prikazuje drug pristop, prva vrstica prikazuje originalno sliko.



Slika 4.5: Slike iz testne množice, ki so bile obarvane z k pristopi opisanimi v tem delu in pristopi iz sorodnih del. Vsaka vrstica prikazuje drug pristop, prva vrstica prikazuje originalno sliko.

globalno mrežo zelo majhno razliko v napaki, pri barvanju slik večjih velikosti glede na napako na osnovni velikosti, medtem ko je ta razlika pri pristopu Iizuka in sod. merjena v RMSE kar 6.18.

Tabela 4.2: Primerjava napak pristopov Iizuka in sod. ter *regresija po delih* z *globalno mrežo* pri barvanju dveh velikosti slik. 224×224 je velikost na kateri je bila mreža naučena, druga velikost je uporabljena za testiranje razlike v barvanju večjih slik. Za računaje napake so uporabljene metrike opisane v poglavaju 3.8.

Pristop	Velikost slik	RMSE	PSNR
Iizuka in sod.	224	10.018	24.750
	896	16.136	20.906
Reg. po delih	224	9.892	24.700
	896	10.096	24.523

Poglavlje 5

Zaključek

Dodatek A

Spearmanova korelacija rangov med metodami

*DODATEK A. SPEARMANOVA KORELACIJA RANGOV MED
METODAMI*

Tabela A.1: Tabela prikazuje vrednosti Spearmanove korelacije rangov med pristopi opisanimi v tem delu. Podrobnosti so opisane v poglavju 4.1.

	Zang in sod.	Iizuka in sod.	Dahl	Reg. po delih	Reg. po delih - brez softmax	Reg. po delih - brez globalne	Reg. celotna slika	Reg. celotna - brez globalne	Reg. celotna VGG	Klas. brez uteži - plitva arh.	Klas. brez uteži - globja arh.	Klas. z utežmi globja arh.	Klas. z utežmi - plitva arh.
Zang	1.00	0.86	0.86	0.86	0.84	0.88	0.87	0.88	0.89	0.90	0.94	0.92	0.90
Iizuka	0.86	1.00	0.89	0.94	0.94	0.90	0.94	0.92	0.93	0.90	0.85	0.85	0.88
Dahl	0.86	0.89	1.00	0.90	0.90	0.98	0.88	0.95	0.91	0.86	0.88	0.87	0.84
R. del.	0.86	0.94	0.90	1.00	0.94	0.90	0.93	0.91	0.93	0.90	0.86	0.85	0.88
R. del. - brez sm.	0.84	0.94	0.90	0.94	1.00	0.91	0.93	0.91	0.92	0.87	0.84	0.83	0.86
R. del. - brez gl.	0.88	0.90	0.98	0.90	0.91	1.00	0.90	0.96	0.92	0.87	0.89	0.88	0.85
R. cel.	0.87	0.94	0.88	0.93	0.93	0.90	1.00	0.91	0.93	0.90	0.86	0.86	0.89
R. cel. - brez gl.	0.88	0.92	0.95	0.91	0.91	0.96	0.91	1.00	0.93	0.87	0.86	0.85	0.85
R. cel. - VGG	0.89	0.93	0.91	0.93	0.92	0.92	0.93	0.93	1.00	0.90	0.88	0.88	0.89
K. brez ut. - pl.	0.90	0.90	0.86	0.90	0.87	0.87	0.90	0.87	0.90	1.00	0.92	0.92	0.94
K. brez ut. - gl.	0.94	0.85	0.88	0.86	0.84	0.89	0.86	0.86	0.88	0.92	1.00	0.98	0.91
K. ut. - gl.	0.92	0.85	0.87	0.85	0.83	0.88	0.86	0.85	0.88	0.92	0.98	1.00	0.92
K. ut. - pl.	0.90	0.88	0.84	0.88	0.86	0.85	0.89	0.85	0.89	0.94	0.91	0.92	1.00

Dodatek B

Podrobnosti arhitektur

Tabela B.1: Tabela prikazuje nivoje plitve arhitekture z globalno mrežo in njihove parametre. K predstavlja število kanalov izhoda nevronске mreže, J pove velikost jedra, Ko je korak (ang. *stride*) uporabljen na nivoju in Akt predstavlja aktivacijo po vsakem nivoju. Zadnji konvolucijski nivo nima definiranega števila izhodnih kanalov, saj je to odvisno od pristopa, ki smo ga uporabili. Pri nadzorčenju smo vedno nadzorčili s faktorjem 2, kar pomeni, da ima tensor, ki predstavlja izhod po širini in višini dvakratno velikost.

Nivo	K	J	K	Akt
Vhod	1	-	-	-
2D konvolucija	64	3	2	Relu
2D konvolucija	128	3	1	Relu
2D konvolucija	128	3	2	Relu
2D konvolucija	256	3	1	Relu
2D konvolucija	256	3	2	Relu
2D konvolucija	512	3	1	Relu
2D konvolucija	512	3	1	Relu
2D konvolucija	256	3	1	Relu
Združitev z globalno mrežo	512	-	-	-
2D konvolucija	256	3	1	Relu
Nadvzorčenje	256	-	-	-
2D konvolucija	256	3	1	Relu
2D konvolucija	256	3	1	Relu
Nadvzorčenje	256	-	-	-
2D konvolucija		3	1	Relu

Tabela B.2: Tabela prikazuje nivoje globje arhitekture z globalno mrežo in njihove parametre. K predstavlja število kanalov izhoda nevronske mreže, J pove velikost jedra, Ko je korak (ang. *stride*) uporabljen na nivoju, Akt predstavlja aktivacijo po vsakem nivoju in Reg določa stopnjo regularizacije. V vseh nivojih uporabljamo $L2$ regulrizacijo [42]. Zadnji konvolucijski nivo nima definiranega števila izhodnih kanalov, saj je to odvisno od pristopa, ki smo ga uporabili.

Zap. št.	Nivo	K	J	Ko	Akt	Reg
0	Vhod	1	-	-	-	-
1	2D konvolucija	64	3	1	Relu	0.01
2	Maks. združevanje	64	2	2	-	-
3	2D konvolucija	128	3	1	Relu	0.01
4	2D konvolucija	128	3	1	Relu	0.01
5	2D konvolucija	128	3	1	Relu	0.01
6	Vsota z 3	128	-	-	-	-
7	2D konvolucija	128	3	1	Relu	0.01
8	Maks. združevanje	128	2	2	-	-
9	2D konvolucija	256	3	1	Relu	0.01
10	2D konvolucija	256	3	1	Relu	0.01
11	2D konvolucija	256	3	1	Relu	0.01
12	Vsota z 9	256	-	-	-	-
13	2D konvolucija	256	3	1	Relu	0.01
14	Maks. združevanje	265	2	2	-	-
15	2D konvolucija	512	3	1	Relu	0.01
16	2D konvolucija	512	3	1	Relu	0.01
17	2D konvolucija	512	3	1	Relu	0.01
18	Vsota z 15	512	-	-	-	-
19	2D konvolucija	512	3	1	Relu	0.01
20	2D konvolucija	256	3	1	Relu	0.01
21	Združitev z globalno mrežo	512	-	-	-	-
22	2D konvolucija	128	3	1	Relu	0.01

23	2D transponirana kovnolucija	64	3	2	Relu	0.01
24	2D kovnolucija	64	3	1	Relu	0.01
25	2D kovnolucija	64	3	1	Relu	0.01
26	2D transponirana kovnolucija	64	3	2	Relu	0.01
27	2D kovnolucija	32	3	1	Relu	0.01
28	2D kovnolucija		3	1	Relu	0.01

Tabela B.3: Tabela prikazuje nivoje dopolnjene VGG mreže in njihove parametre. K predstavlja število kanalov izhoda nevronске mreže, J pove velikost jedra, Ko je korak (ang. *stride*) uporabljen na nivoju, Akt predstavlja aktivacijo po vsakem nivoju in Reg določa stopnjo regularizacije. V vseh nivojih uporabljamo $L2$ regulirzacijo [42]. Zadnji konvolucijski nivo nima definiranega števila izhodnih kanalov, saj je to odvisno od pristopa, ki smo ga uporabili. Vsako nadzorčenje prostorske dimenzije poveča za 2 krat.

Zap. št.	Nivo	K	J	Ko	Akt	Reg
0	Vhod	1	-	-	-	-
1	VGG-16 mreža	512	-	-	-	-
2	Nadvzorčenje	512	-	-	-	-
3	2D konvolucija	256	3	1	Relu	0.01
4	2D konvolucija	256	3	1	Relu	0.01
5	Nadvzorčenje	256	-	-	-	-
6	2D konvolucija	128	3	1	Relu	0.01
7	2D konvolucija	128	3	1	Relu	0.01
8	Nadvzorčenje	128	-	-	-	-
9	2D konvolucija	64	3	1	Relu	0.01
10	2D konvolucija	164	3	1	Relu	0.01
11	Nadvzorčenje	64	-	-	-	-
12	2D konvolucija	32	3	1	Relu	0.01
13	2D konvolucija		3	1	Relu	0.01

Literatura

- [1] M. ARCHAMBAULT, “A Brief History of Color Photography, From Dream to Reality.” <https://petapixel.com/2015/10/11/a-brief-history-of-color-photography-from-dream-to-reality/>. Dostopano: 28. 6. 2017.
- [2] A. Levin, D. Lischinski, in Y. Weiss, “Colorization using optimization,” v *ACM Transactions on Graphics (TOG)*, vol. 23, pp. 689–694, ACM, 2004.
- [3] Y.-C. Huang, Y.-S. Tung, J.-C. Chen, S.-W. Wang, in J.-L. Wu, “An adaptive edge detection based colorization algorithm and its applications,” v *Proceedings of the 13th annual ACM international conference on Multimedia*, pp. 351–354, ACM, 2005.
- [4] M. Koleini, S. A. Mobadjemi, in P. Moallem, “Automatic Black and White Film Colorization Using Texture Features and Artificial Neural Networks,” *Journal of the Chinese Institute of Engineers*, vol. 33, no. 7, pp. 1049–1057, 2010.
- [5] P. Shirley, “Color transfer between images,” *IEEE Corn*, vol. 21, pp. 34–41, 2001.
- [6] Y.-W. Tai, J. Jia, in C.-K. Tang, “Local color transfer via probabilistic segmentation by expectation-maximization,” v *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, vol. 1, pp. 747–754, IEEE, 2005.

- [7] Z. Cheng, Q. Yang, in B. Sheng, “Deep Colorization,” v *The IEEE International Conference on Computer Vision (ICCV)*, dec 2015.
- [8] A. Deshpande, J. Rock, in D. Forsyth, “Learning large-scale automatic image colorization,” v *Proceedings of the IEEE International Conference on Computer Vision*, vol. 11-18-Dece, pp. 567–575, 2016.
- [9] R. Dahl, “Automatic Colorization.” <http://tinyclouds.org/colorize/>, 2016. Dostopano: 1. 6. 2017.
- [10] K. Simonyan in A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” *CoPR*, vol. abs/1409.1, sep 2014.
- [11] R. Zhang, P. Isola, in A. A. Efros, “Colorful Image Colorization,” v *Computer Vision – ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III* (B. Leibe, J. Matas, N. Sebe, in M. Welling, eds.), pp. 649–666, Cham: Springer International Publishing, 2016.
- [12] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, in L. Fei-Fei, “ImageNet Large Scale Visual Recognition Challenge,” *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.
- [13] G. Larsson, M. Maire, in G. Shakhnarovich, “Learning Representations for Automatic Colorization,” *arXiv preprint arXiv:1603.06668*, 2016.
- [14] S. Iizuka, Edgar Simo-Serra, in H. Ishikawa, “Let there be Color!: Joint End-to-end Learning of Global and Local Image Priors for Automatic Image Colorization with Simultaneous Classification,” v *SIGGRAPH ’16*, vol. 35, p. 110, ACM, 2016.
- [15] A. Gibson, C. Nicholson, in J. Patterson, “Introduction to Deep Neural Networks.” <https://deeplearning4j.org/neuralnet-overview>. Dostopano: 26. 6. 2017.

- [16] A. Karpathy, “Neural Networks Part 1: Setting up the Architecture.” CS231n Convolutional Neural Networks for Visual Recognition, Stanford University, 2016.
- [17] R. Collobert in J. Weston, “A unified architecture for natural language processing: Deep neural networks with multitask learning,” v *Proceedings of the 25th international conference on Machine learning*, pp. 160–167, ACM, 2008.
- [18] A. Karpathy, “Understanding and Visualizing Convolutional Neural Networks.” CS231n Convolutional Neural Networks for Visual Recognition, Stanford University, 2016.
- [19] Y. LeCun, Y. Bengio, in Others, “Convolutional networks for images, speech, and time series,” *The handbook of brain theory and neural networks*, vol. 3361, no. 10, p. 1995, 1995.
- [20] A. Krizhevsky, I. Sutskever, in G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,” v *Advances in Neural Information Processing Systems 25* (F. Pereira, C. J. C. Burges, L. Bottou, in K. Q. Weinberger, eds.), pp. 1097–1105, Curran Associates, Inc., 2012.
- [21] P. Nishad in R. Manicka Chezian, “Various Colour Spaces and Colour Space Conversion,” *Journal of Global Research in Computer Science*, vol. 4, no. 1, pp. 44–48, 2013.
- [22] L. Prangnell, “Visible Light-Based Human Visual System Conceptual Model,” *CoRR*, vol. abs/1609.0, 2016.
- [23] S. Bansal in D. Aggarwal, “Color Image Segmentation Using CIELab Color Space Using Ant Colony Optimization,” *IJCSET*, vol. 1, no. 7, pp. 415–420, 2011.
- [24] K. Jack, “Color spaces,” v *Video Demystified - A Handbook for the Digital Engineer*, ch. Color spac, pp. 15–34, Elsevier, 1989.

- [25] I. L. Weatherall in B. D. Coombs, “Skin Color Measurements in Terms of CIELAB Color Space Values,” *Journal of Investigative Dermatology*, vol. 99, pp. 468–473, oct 1992.
- [26] H. Everding, “CIELAB boundaries – the borders of the CIE-LAB color space.” <https://www.freiefarbe.de/en/grenzen-des-cielab-farbraums/>. Dostopano: 27. 6. 2017.
- [27] C. Connolly in T. Fleiss, “A study of efficiency and accuracy in the transformation from RGB to CIELAB color space,” *IEEE Transactions on Image Processing*, vol. 6, pp. 1046–1048, jul 1997.
- [28] N. Ohta in A. R. Robertson, “CIE Standard Colometric System,” v *Colorimetry: Fundamentals and Applications*, pp. 63–114, 2006.
- [29] B. J. Lindbloom, “XYZ to Lab.” http://www.brucelindbloom.com/index.html?Eqn_XYZ_to_Lab.html, 2016. Dostopano: 26. 6. 2017.
- [30] V. Dumoulin in F. Visin, “A guide to convolution arithmetic for deep learning,” *Xiv preprint arXiv:1603.07285*, mar 2016.
- [31] S. Wu, S. Zhong, in Y. Liu, “Deep residual learning for image steganalysis,” *Multimedia Tools and Applications*, 2017.
- [32] J. M. Joyce, “Kullback-leibler divergence,” v *International Encyclopedia of Statistical Science*, pp. 720–722, Springer, 2011.
- [33] S. Mannor, D. Peleg, in R. Rubinstein, “The cross entropy method for classification,” v *Proceedings of the 22nd international conference on Machine learning - ICML '05*, (New York, New York, USA), pp. 561–568, ACM Press, 2005.
- [34] D. P. Kingma in J. Ba, “Adam: A Method for Stochastic Optimization,” *CoRR*, vol. abs/1412.6, 2014.

- [35] Y. Ke in R. Sukthankar, "PCA-SIFT: A more distinctive representation for local image descriptors," v *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, vol. 2, pp. II–506–II–513, IEEE, 2004.
- [36] H. Bay, T. Tuytelaars, in L. Van Gool, "SURF: Speeded up robust features," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 3951 LNCS, pp. 404–417, 2006.
- [37] D. Saupe, R. Hamzaoui, in H. Hartenstein, "Fractal Image Compression - An Introductory Overview," tech. rep., 1997.
- [38] S. T. Welstead, *Fractal and wavelet image compression techniques*, vol. 40. Spie Press, 1999.
- [39] J. Hauke in T. Kossowski, "Comparison of values of Pearson's and Spearmann's correlation coefficients on the same sets of data," *Quaestiones geographicae*, vol. 30, no. 2, p. 87, 2011.
- [40] F. Wickelmaier, "An introduction to MDS," *Sound Quality Research Unit, Aalborg University, Denmark*, 2003.
- [41] S. Ramaswamy, R. Rastogi, in K. Shim, "Efficient algorithms for mining outliers from large data sets," v *Proceedings of the 2000 ACM SIGMOD international conference on Management of data - SIGMOD '00*, pp. 427–438, 2000.
- [42] D. J. C. MacKay, "A practical Bayesian framework for backpropagation networks," *Neural computation*, vol. 4, no. 3, pp. 448–472, 1992.