

VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
INFORMATIKOS KATEDRA

**Gilaus neuroninio tinklo CenterNet ir žinių
perdavimo tyrimas ortofotografinių vaizdų panašumo
nustatymui**

**Investigation of CenterNet and Transfer Learning for
Similarity Estimation of Aerial Images**

Magistro baigiamasis darbas

Atliko: Justinas Šeibokas (parašas)

Darbo vadovas: Dr. Virginijus Marcinkevičius (parašas)

Recenzentas: (parašas)

Vilnius – 2020

Santrauka

Darbo tikslas pasiūlyti giliojo neuroninio tinklo CenterNet modifikaciją, kuri leidžia efektyviai palyginti du žemėlapius.

Darbe išnagrinėtas gilusis neuroninis tinklas CenterNet, susipažinta su jo architektūra ir jame naudojamais neuroninių tinklų modeliais. Apžvelgti kiti metodai ir neuroniniai tinklai lyginantys vaizdų panašumus. Pasiūlyta CenterNet modifikacija, kurią galima naudoti vaizdų panašumams lyginti. Susipažinta su metrikomis skirtomis matuoti vaizdų panašumams. Metrikos naudotos eksperimentuose nustatyti neuroninio tinklo modelių rezultatų tikslumui. Apžvelgta mokymosi perdavimo būdai ir atliktas eksperimentas, kurie tinklo svoriais geriausiai tinka pasiūlyto tinklo mokymuisi. Apžvelgtos trys tripleto tinklų tikslo funkcijos ir atlikti eksperimentai, kurie padėjo nustatyti efektyviausias funkcijas siūlomui tinklo modeliui mokinti. Sugeneruotas duomenų rinkinys skirtas ortofotografinių duomenų panašumui tirti.

Pasiūlytas modifikuotas neuroninio tinklo CenterNet modelis, pagal apžvelgtas metrikas su sugeneruotu duomenų rinkiniu savo tikslumu pralenkia kitą gilųjį neuroninį tinklą skirtą ortofotografinių vaizdų vienodumo nustatymui, kuris yra publikuotas moksliniame žurnale.

Raktažodžiai: neuroniniai tinklai, CenterNet, ortofotografija, vaizdų panašumai, tripleto tinklai, tripleto tikslo funkcijos, mokymosi perdavimas.

Summary

The aim of the work is to propose deep neural network CenterNet modification capable to effectively compare two aerial maps.

This paper presents an investigation of CenterNet neural network and neural networks inside him and an investigation of other image similarity methods and neural networks that compare them. Proposed modification of CenterNet for aerial image similarity. Investigated learning transfer for neural networks to learn aerial image similarity. Investigated three triplet loss functions for learning aerial image similarity. Different similarity metrics are investigated and used to measure proposed neural network model accuracies. An aerial image dataset was created to perform experiments with triplet networks.

Proposed neural network model for aerial image similarity, outperforms existing neural network model that is published in science journal, while using created data set and investigated similarity measures.

Keyword: neural networks, CenterNet, aerial Images, image similarity, triplet networks, triplet loss, learning transfer.

Turinys

1. Įvadas	4
2. Gilūs dirbtiniai neuroniniai tinklai ir vaizdų analizė	6
2.1. Gilūs dirbtiniai neuroniniai tinklai	6
2.2. CenterNet tinklas	7
2.3. CenterNet baziniai tinklai	9
1.1. CenterNet rezultatai	11
1.2. Vaizdų panašumas	13
1.3. Tripleto tikslo funkcijos	15
1.4. Tripleto tinklo architektūra	17
1.5. Mokymo perdavimas	20
1.6. CenterNet išankstinis mokymas	23
1.7. Konvolucijos	24
1.8. Metrikos	28
1.8.1. Tikslumas	28
1.8.2. Vidutinis tikslumas	29
1.8.3. Pearsono koreliacija	30
2. Eksperimentai	31
2.1. Duomenų rinkinys	31
2.2. Tinklo mokymas	31
2.3. Eksperimentai su išankstiniu mokymu	33
2.3.1. Tarpinis mokymas	34
2.3.2. Rezultatai	34
2.3.2.1. Tikslumas	34
2.3.2.2. Vidutinis tikslumas	35
2.3.2.3. Pearsono koreliacija	36
2.4. Eksperimentai su tripleto tikslo funkcijomis	39
2.4.1. Rezultatai	40
2.4.1.1. Tikslumas	40
2.4.1.2. Vidutinis tikslumas	41
2.4.1.3. Pearsono koreliacija	42
2.5. Palyginimas su VGG16	45
3. Išvados	47
Šaltinių sąrašas	50

1. Įvadas

Bepiločiai orlaiviai (*angl. Unmanned Aerial Vehicle (UAV)*) jau dabar yra ganėtinai plačiai naudojami. Naudojami ne tik kariniams tikslams, bet ir filmavimams, nuotoliniam stebėjimui, gaisrų prevencijai, žmogaus sveikatai žalingų, ar neprieinamų vietų žvalgybai, kaip pastatų griuvėsiai ar tiesiog pramogai. Galima naudoti kaip pirmuosius gelbėtojus greitam nukentėjusiųjų suradimui po stichinių nelaimių, transportuoti krovinius, purkšti laukus trąšomis ir daugeliui kitų.

Išvardintas veiklas atlikti galima ir dabar, tačiau šiuo metu bepiločiai orlaiviai dažniausiai yra valdomi žmonių. Automatizavus bepiločius orlaivius galima būtų sumažinti jų eksploatacines išlaidas, bei supaprastinti naudojimą. Vienas iš pagrindinių šio uždavinio sunkumų yra orlaivio navigacija aplinkose, kur nėra ryšio su navigacijos palydovu. Tradicinės autopiloto sistemos nesugeba sėkmingai naviguoti, kai GPS signalas yra užblokuojamas, prarastas arba tiesiog negalimas. Orlaivis turi mokėti nustatyti savo pozicija nepasikliaudamas išoriniais signalais. Tam galima naudoti vizualią odometriją (*angl. visual odometry*) ar žemėlapių sudarymą ir lokalizaciją (*angl. simultaneous localization and mapping*) – SLAM, naudojant vaizdo kamerą esančią orlaivyje. Pastarieji būdai duoda labai gerus rezultatus uždarose patalpose ar skrendant arti žemės (< 100 metrų). SLAM ir vizualiai odometrijai nereikia žinoti aplinkos žemėlapių, kurioje jie naudojami, toki būdai vadinama nereikalaujantys žemėlapių (*angl. map-less*). Abu šitie metodai linkę generuoti klaidas, kai skrydžio distancija yra didesnė nei vienas kilometras ar skrydžio trajektorija yra aukštesnė nei 100 metrų. Žemėlapius naudojančios sistemos gali padidinti tikslumą naviguojant ilgų skrydžių distancijose, palyginus su nereikalaujančiais žemėlapių būdais. Tokios sistemos padaro nuotrauka su orlaivyje esančia kamera ir palygina ją su turimu ortofotografiniu žemėlapiu, ieškant kuo panašesnės vietos ir taip bandant nustatyti savo padėtį žemėlapyje.

Palyginti du paveikslėlius galima lyginant jų pikselius (*angl. pixel to pixel*). Visus ar atsitiktinai pasirinktų vietų. Tačiau toks būdas geriausiai tinka tik labai panašioms vaizdams lyginti, kai vaizdų fotografavimo sąlygos yra identiškos arba labai panašios. Turėti žemėlapius, kurie atitiktų visus metų laikus, apšvietimo sąlygas ir atspindėtų realią situaciją yra praktiškai neįmanoma užduotis.

Kitas būdas vietoje to, kad tikrinti visus ar atsitiktinius pikselius yra lyginti vaizdo savybes. Vienos vietos paveikslėliuose savyje turi daugiau informacijos, nei kitos (pvz. žemėlapyje pievoje esantis statinys gali suteikti daugiau informacijos, nei tiesiog tuščia pieva). Tokių vietų parinkimas vadinamas – raktinių taškų suradimu (*angl. keypoint extraction*). Pasinaudojus raktinių taškų vienodumo atitikimu (*angl. keypoint matching*) galima įvertinti vaizdų panašumą [Tya19]. Didžiausias tokio metodo trūkumas yra, kad didėjant raktinių taškų ir vaizdų skaičiui labai greitai prastėja algoritmo greitimeika.

Paveikslėlių galima išreikšti vektorių pavidalu, kur kiekviena vertė atitinka vaizdo pikselio vertę. Turint tokią skaitinę vaizdo išraišką galima apskaičiuoti vaizdų panašumą naudojant kokią nors atstumo metriką. Spalvoti vaizdai dažniausiai išreiškiami trimis ar keturiomis matricomis, kas apsunkina panašumo skaičiavimą. Todėl prieš skaičiavimą vaizdą galima paversti į nespalvotą (*angl. grayscale*) norint sumažinti matricų skaičių, bet taip prarandama ir dalis informacijos. Kad išvengtų šios problemos galima naudoti gilųjį neuroninį tinklą paskaičiuoti paveikslėlio požymių vektoriams (*angl. embeddings*), kurie bus lyginami su vaizdų panašumo metrika, o ne tiesioginės pikselių vertės.

Pastaruoju metu, gilieji neuroniniai tinklai (ResNet [HYS+15], DLA [YWS+19], Inception [SLJ+14], CenterNet [ZWK19]) demonstruoja labai gerus rezultatus vaizdų klasifikavime. Vaizdų klasifikavimas yra artimas uždavinys vaizdų panašumo nustatymui, todėl būtų galima panaudoti jau egzistuojančius tinklo modelius ir vaizdų panašumui lyginti. Patiems tinklams apmokyti reikalingi labai dideli duomenų kiekiai, todėl pasinaudojus mokymo perdavimu (*angl. transfer learning*), galima imti jau apmokytą gilųjį neuroninį tinklą ir panaudoti tokio tinklo bazinius sluoksnius vaizdo savybių išgavimui, laikant, kad jis jau pakankamai gerai moka atpažinti vaizdo savybes. Papildomai pridėjus kelis mokomus sluoksnius juos apmokyti naudojant tripleto tikslo funkciją skirta vaizdų panašumui mokymui (FaceNet [SKP15], VGG16 [VMR20]).

Darbo tikslas pasiūlyti giliojo neuroninio tinklo CenterNet modifikaciją, kuri leidžia efektyviai palyginti du žemėlapius. Darbe planuojami atlikti uždaviniai:

- susipažinti su CenterNet tinklu ir jo architektūra, kad būtų galima ją efektyviai modifikuoti vaizdų vienodumo lyginimui. Susipažinti esančiais giliųjų neuroninių tinklų sprendimais vaizdų vienodumui lyginti ir apžvelgti esamas metrikas tokio tipo tinklų efektyvumui įvertinti, kurias galima naudoti eksperimentų dalyje,
- pasiūlyti CenterNet tinklo modifikaciją, kuri tiktų vaizdų panašumui apskaičiuoti,
- atlikti eksperimentus, kaip pradiniai svoriai gali įtakoti tinklo mokymąsi vaizdų panašumui atpažinti ir juos įvertinti naudojant nustatytas metrikas,
- atlikti eksperimentus, kokią įtaką turi skirtingos tripleto tikslo funkcijos, tinklo mokymuisi vaizdų panašumui atpažinti ir jas įvertinti naudojant nustatytas metrikas.

CenterNet[ZWK19] tinklas yra sudarytas iš kelių modifikuotų giliųjų neuroninių tinklų (Hourglass-114 [NYD+16], DLA-34 [YWS+19], ResNet-18 [HYS+15] ir ResNet-101[HYS+15]), kurie apmokyt naudojami autorių sukurtą objektų detektorius. Todėl jis ir buvo pasirinktas šiam darbui, taip užtikrinant vienodumą tarp tiriamų tinklų, kadangi visi buvo apmokomi naudojant tą pačią sistemą.

Darbe lyginami trys gilieji neuroniniai tinklai (DLA-34, ResNet-18, ResNet-101) esantys CenterNet tinkle. Pasiūloma architektūra, kaip pritaikyti tokius tinklus vaizdų panašumui atpažinti. Architektūra pasiūlyta įvertinus kitus tinklus vaizdų vienodumui atpažinti ir esamą CenterNet architektūrą.

Tinklui įvertinti apžvelgiamos ir parenkamos metrikos skirtos vaizdų panašumui vertinti. Kadangi tinklams mokinti reikalingi labai dideli sužymėtų duomenų kiekiai, naudojama mokymosi perdavimas (*angl. transfer learning*) tinklui apmokinti. Atliekamas eksperimentas, kaip skirtingi pradiniai tinko svoriai įtakoja tinko rezultatus. Rezultatai palyginami nustatytais vaizdų panašumo metrikomis.

Viena svarbiausių dalių giliojo neuroninio tinklo mokyme yra tikslo funkcija. Darbe apžvelgiamos trys skirtingos tripleto tikslo funkcijos, kurios yra naudojamos neuroniniuose tinkluose vaizdų panašumui nustatyti. Atliekami eksperimentai, kaip tokios tikslo funkcijos įtakoja tinklo rezultatus ortofotografinių vaizdų panašumo mokyme. Rezultatai palyginami nustatytais vaizdų panašumo metrikomis.

2. Gilūs dirbtiniai neuroniniai tinklai ir vaizdų analizė

2.1. Gilūs dirbtiniai neuroniniai tinklai

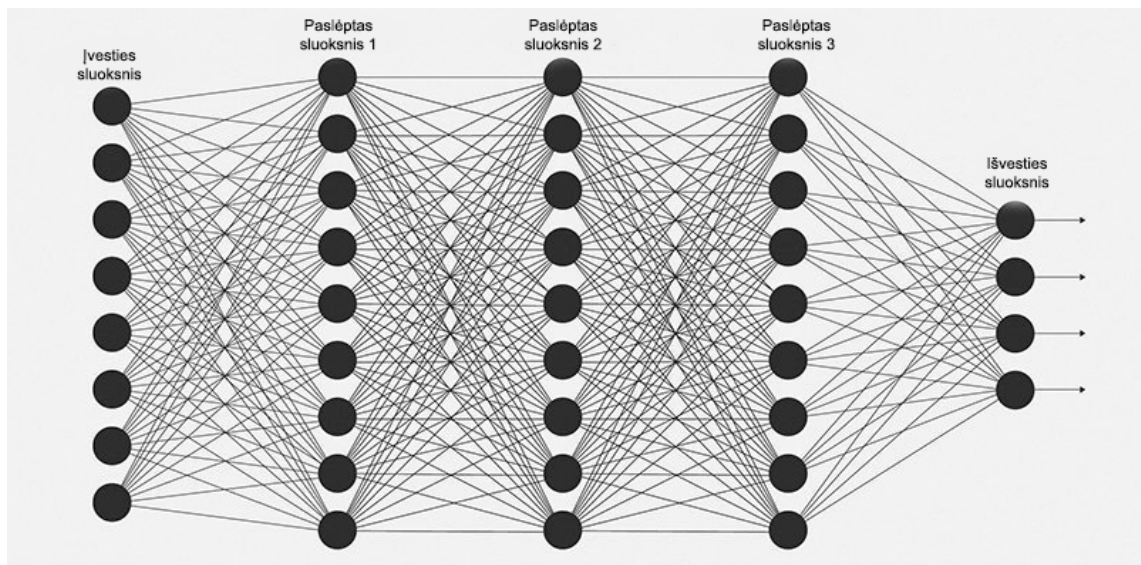
Dirbtiniai neuroniniai tinklai dabar turbūt yra populiariausia informatikos mokslų tyrimo sritis. Tokio populiarumo jie sulaukė dėl pakankamai gerų rezultatų įvairiuose tyrimų srityse, aplenkiant tikslumu kitus iki tol naudotus metodus. Tinklai buvo sukurti bandant mėgdžioti biologinio neurono veikimo principą. Toki tinklai “išmoksta” spręsti uždavinius analizuodami pavydžius, o ne yra programuoti atlikti jiems.

Toks tinklas yra sudarytas iš sujungtų dirbtinių neuronų, kurie priklauso kažkuriam iš sluoksnių. Tinklas paprastai turi vieną išvesties, vieną įvesties sluoksnį ir kintamą skaičių paslėptų sluoksnių. Kiekvienas neuronas tinkle gali būti sujungtas su daugybe kitų neuronų, priklausančių kitam sluoksniui. Per tuos sujungimus yra perduodama informacija. Bendra giliojo neuroninio tinklo schema matoma 1 paveikslėlyje. Pagrindinis skirtumas tarp paprasto neuroninio tinklo ir giliojo neuroninio tinklo yra tai, kad jis turi daugiau nei vieną paslėptą sluoksnį.

Įvesties sluoksnyje pateikiama informacija tinklui. Neuronas atlieka skaičiavimus su duomenimis ir juos perduoda kitam sluoksniui. Sekantis sluoksnis jau nemato pradinės informacijos ir atlieka skaičiavimus su ta, kuria gavo. Viskas kartojama kol pasiekiami išvesties sluoksniai, tai ir bus neuroninio tinklo rezultatas.

Gilieji neuroniniai tinklai naudojami atlikti įvairiomis užduotims [Mit17]:

- autonominiams automobiliams (*angl. self-driving cars*) valdyti,
- vaizdų atpažinimui,
- ligų diagnostikai,
- paieškai balsu, balsu valdomiems įrenginiams,
- automatiniam vertimui,
- prognozuoti žemės drebėjimams,
- prognozuoti ekonomikos svyravimams,
- reklamai,
- tyliųjų kino filmų garo takelių generavimui,
- nespaltvotų vaizdų (nuotraukų, filmų) spalvinimui.



pav. 1 Bendra gilaus neuroninio tinklo schema.¹

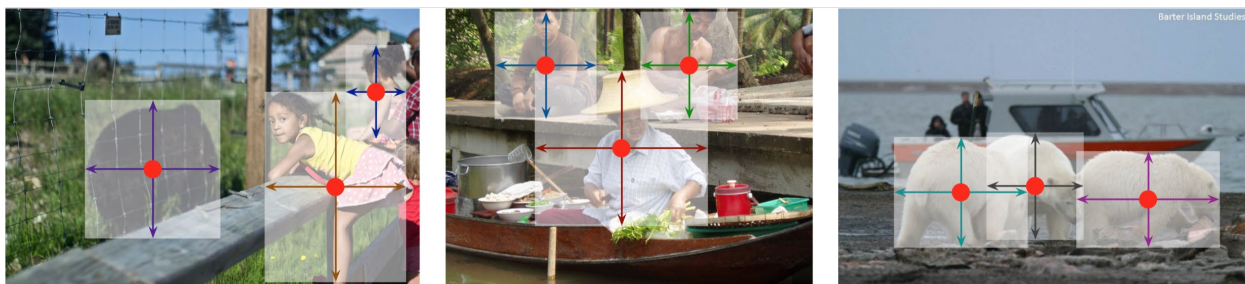
Vieni populiariausiu giliųjų neuroninių tinklų yra konvoliuciniai neuroniniai tinklai, skirti vaizdų atpažinimui, naudojantys konvoliucijas išgauti vaizdo savybėms. Plačiau apie konvoliucijas aprašyta 2.10. skyriuje. Tiriamas CenterNet tinklas priklauso konvoliuciniams tinklams.

2.2. CenterNet tinklas

CenterNet [ZWK19] nėra vienos konkrečios architektūros gilusis dirbtinis neuroninis tinklas. Tai sistema savyje turinti kelis skirtingos architektūros giliuosius konvoliucinius neuroninius tinklus. Kurie naudoja autorių sukurtą objektų aptikimo detektorių vaizduose.

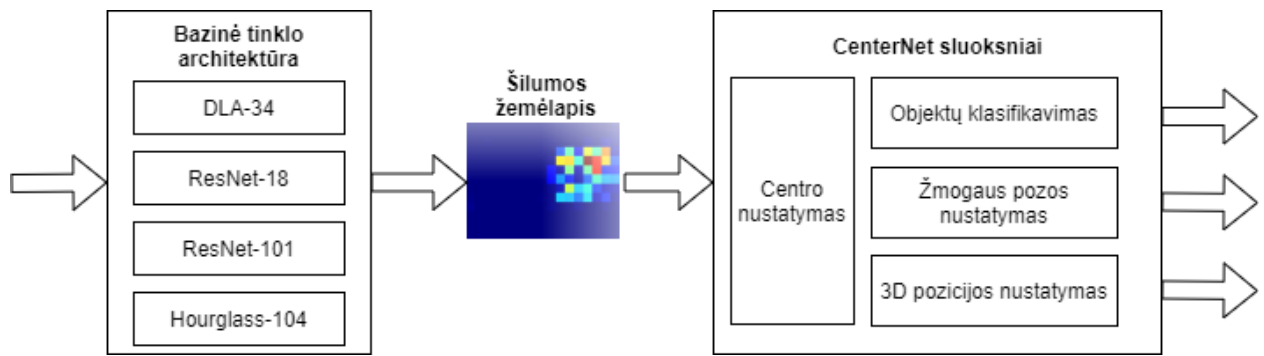
¹ Šaltinis: <https://towardsdatascience.com/training-deep-neural-networks-9fdb1964b964>

Kitaip nei kiti objektų identifikavimo tinklai, kurie nustatinėja visas potencialias objekto pozicijas vaizde ir jas klasifikuoja, CenterNet rasdamas raktinius taškus (*angl. keypoint estimation*) aptinka tik objekto centro tašką. Ir vėliau remdamasis juo suranda kitas objekto savybes, tokias kaip 3D lokacija, orientacija ar poza. Toks sprendimas ieškoti centro taško lėmė, kad CenterNet yra greitesnis ir nedaug nusileidžiantis tikslumu už tuo metu buvusius pažangiausius (*angl. state of the art*) panašaus tipo objektų aptikimo tinklus [ZWK19]. Paveikslėlyje nr. 2 pavaizduota CenterNet objektų centrų aptikimo vizualizacija.



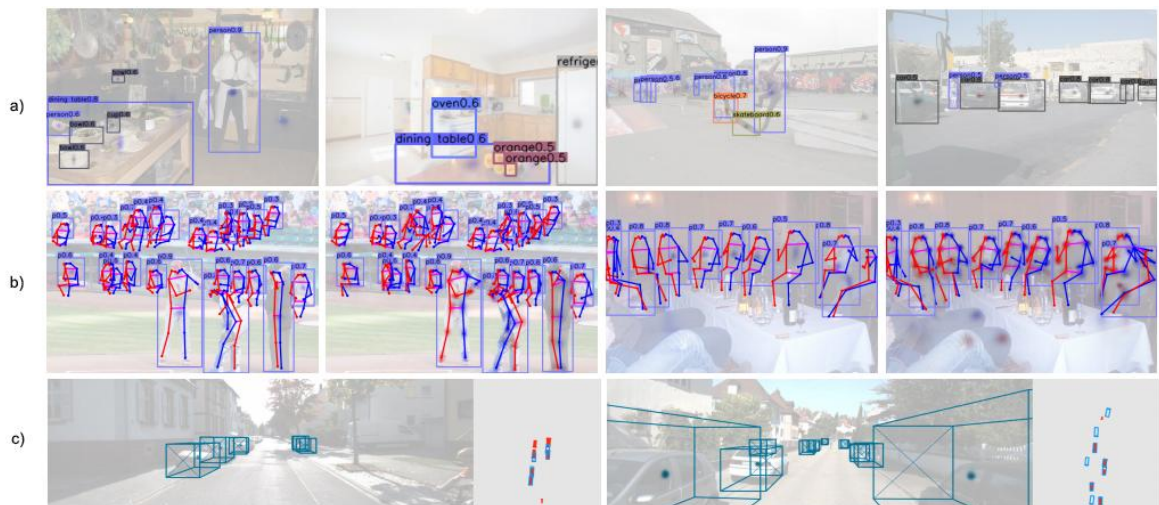
2 pav. CenterNet objektų centrų aptikimas. Šaltinis [ZWK19].

Pirminis CenterNet tinklo uždavinys yra aptikti objekto esminius taškus (*angl. keypoint estimation*), pagal kuriuos ir randamas objektas. CenterNet nėra pirmasis tinklas naudojantis esminių taškų nustatymą objekto aptikimui. Panašiai veikia CornerNet [LD18] ir ExtremeNet [ZZK19]. Įvesties paveikslėlis (*angl. input*) yra paduodamas konvoliuciniam neuroniniam tinklui, kuris generuoja šilumos žemėlapius (*angl. heatmaps*). Toki tinklai pasiūlyti kitų mokslininkų, kurie transformavo pilnai sujungtus (*angl. fully connected*) sluoksnius į konvoliucinius, taip įgalindami neuroninį tinklą išvesti šilumos žemėlapius [LSD15]. Tokiuose šilumos žemėlapiuose “karščiausi” taškai nurodo objekto centrą. Pagal paveikslėlio savybės tame taške, yra nusprendžiama objektą ribojančio stačiakampio (*angl. bounding box*) aukštis ir plotis. Tokį objektų atpažinimo algoritmą galima lengvai pritaikyti atlikti ir kitiems uždaviniams. CenterNet tinklo autoriai taip eksperimentuoja su objektų 3D pozicijų nustatymu (*angl. 3D pose*), bei daugelio žmonių pozų (*angl. multi-person human pose*) atpažinimu. Aprašyta bazinė CenterNet tinklo architektūra pavaizduota paveikslėlyje nr. 3.



3 pav. Bazinė CenterNet tinklo architektūra.

Kaip minėta, CenterNet moka atlikti keletą uždavinių - objektų identifikavimas, objekto 3D pozicija ir žmogaus kūno pozos nustatymas. Tinklo gražinami rezultatai pavaizduoti paveikslėlyje nr. 4. Visas šias užduotis atlieka autoriaus pridėti paskutiniai neuroninio tinklo sluoksniai, kol pats tinklo branduolys visiems uždaviniams yra vienodas. Autorius pridėjo du konvoliucijos 3x3 ir 1x1 sluoksnius atskirtus ReLU aktyvacijos funkcija. Plačiau apie konvoliucijas kalbama skyriuje 2.10.



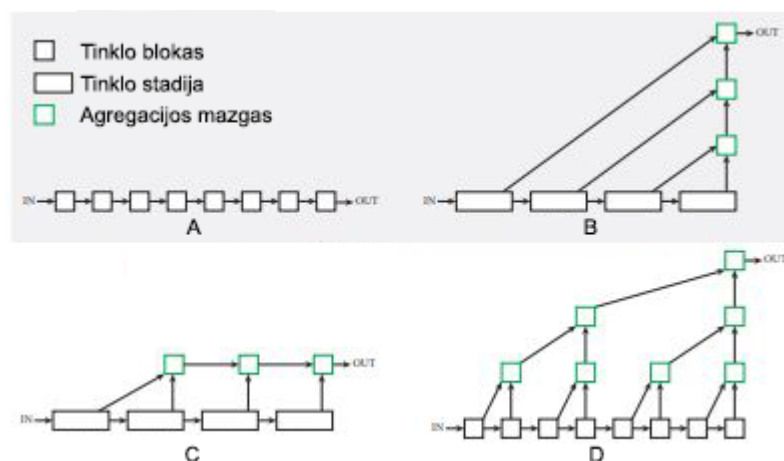
4 pav. CenterNet skirtingų uždavinių rezultato vizualizavimas: a) objektų aptikimas, b) žmogaus pozos aptikimas, c) 3D pozicijos aptikimas. Šaltinis [ZWK19].

2.3. CenterNet baziniai tinklai

CenterNet savyje turi kelis giliuosius neuroninius tinklus: ResNet-18 [HZZ+15], ResNet-101 [HZZ+15], DLA-34 [YWS+19] ir Hourglass-104 [NYD+16]. Paveikslėlyje nr. 3 matyti CenterNet baziniai tinklai. DLA-34 ir ResNet (ResNet-18 ir ResNet-101) tinklai yra modifikuoti. Standartiniai konvoliuciniai sluoksniai juose pakeisti kintančios konvoliucijos (*angl. deformable convolution*) sluoksniais. Apie konvoliucijas plačiau aprašyta skyriuje 2.10.

Hourglass tinklas [NYD+16] arba kaip autorius jį vadina sukrautas (*angl. stacked*) Hourglass tinklas, buvo pristatytas, kaip naujos architektūros konvoliucinis neuroninis tinklas, skirtas žmogaus pozos nustatymui. Tinklo architektūra yra ganėtinai didelės apimties palyginus su kitais tinklais esančiais CenterNet. Taip yra todėl, kad tinklas bando nustatyti vaizdo savybes visuose masteliuose ir jas paskui sujungti į vieną rezultatą, taip gaudamas geresnį vaizdą apie erdvines kūno savybes. FLIC² ir MPII³ etaloniniuose testuose (*angl. benchmarks*) tinklui pavyko pasiekti geriausias rezultatus, pralenkiant visus tuo metu esamus kitus geriausias tinklus [NYD+16]. CenterNet sistemoje šis tinklas taip pat duoda tiksliausias rezultatus, tačiau turi ilgiausią vykdymo laiką (lentelė nr. 1).

DLA-34 [YWS+19] arba Giliųjų Sluoksnių Agregacija (*angl. Deep Layer Aggregation*) Berklio universiteto mokslininkų sukurtas gilusis konvoliucinis neuroninis tinklas. Pagrindinė tinklo naujovė, kaip galima spėti ir iš pavadinimo, yra naujas būdas sujungti vidinius sluoksnius. Autorių sukurtos struktūros pakartotinai ir hierarkiškai sujungia aptiktas savybes, padarydamos tinklus tikslesnius ir turinčius mažiau parametrų. Geriausia tai pavaizduoja paveikslėlis nr. 5. DLA pasiekia geresnius rezultatus lyginant su panašaus sluoksnių kiekio ResNet tinklais, mokant ir testuojant juos su ImageNet⁴ duomenų rinkiniu. CenterNet naudoja šiek tiek modifikuotą DLA-34 tinklą. Praleistos jungtys tarp sluoksnių pakeistos kintamos konvoliucijos sluoksniais iš žemųjų į išvesties sluoksnį. Originalūs konvoliuciniai dimensijų didinimo (*angl. upsampling*) sluoksniai pakeisti kintamos konvoliucijos sluoksniais.



5 pav. A ir B egzistavusios architektūros iki DLA. A - be agregacijos, B - sekli (*angl. shallow*) agregaciją. C ir D autorių pasiūlyti gilios agregacijos sprendimai. Schemos paimtos iš [YWS+19].

ResNet [HZZ+15] arba tinklai su liekana (*angl. Residual Network*) davė didžiulį proveržį kompiuterinės regos, tyrimų srityje. ResNet pagalba galima buvo apmokyti tinklus, kurie turi

² <https://bensapp.github.io/flic-dataset.html>

³ <http://human-pose.mpi-inf.mpg.de/>

⁴ <http://www.image-net.org/>

šimtus ar net tūkstančius sluoksnių ir neaukoti daugybės laiko tam, kaip tai buvo iki ResNet tinklų. Tai davė didelį proveržį objektų ir veidų atpažinime. Pagrindinė tinklo naujovė, kad gradientai sklidimo atgal metodu (*angl. backpropagation*) gali tiesiai pasiekti giliau esančius sluoksnius, neprarasdami savo verčių. Bendra gradiento „peršokimo“ schema pavaizduoti paveikslėlyje nr. 6 b dalyje. CenterNet naudoja modifikuotus ResNet tinklus, gale pridėjus tris konvoliucinius sluoksnius, kurie didina išvesties vaizdo rezoliucija. Taip pat pakeistos visos originalios konvoliucijos kiekviename dimensijų didinimo (*angl. upsampling*) sluoksnyje į kintamas konvoliucijas. ResNet-18 architektūra pavaizduota paveikslėlyje nr. 6 a dalyje. ResNet-101 architektūra yra panaši tiesiog ji turės 101 sluoksnių, ResNet-18 18 sluoksnių. Giliausias ResNet tinklas yra su 151 sluoksnių.

1 lentelė. CenterNet greičio ir tikslumo palyginimas testuojant su COCO⁵ duomenų rinkiniu. Šaltinis [ZWK19].

	Vidutinis tikslumas	Laikas (milisekundės)	Kadrai per sekunde
Hourglass-104	40.3	71	14
DLA-34	37.4	19	52
ResNet101	34.6	22	45
ResNet-18	28.1	7	142

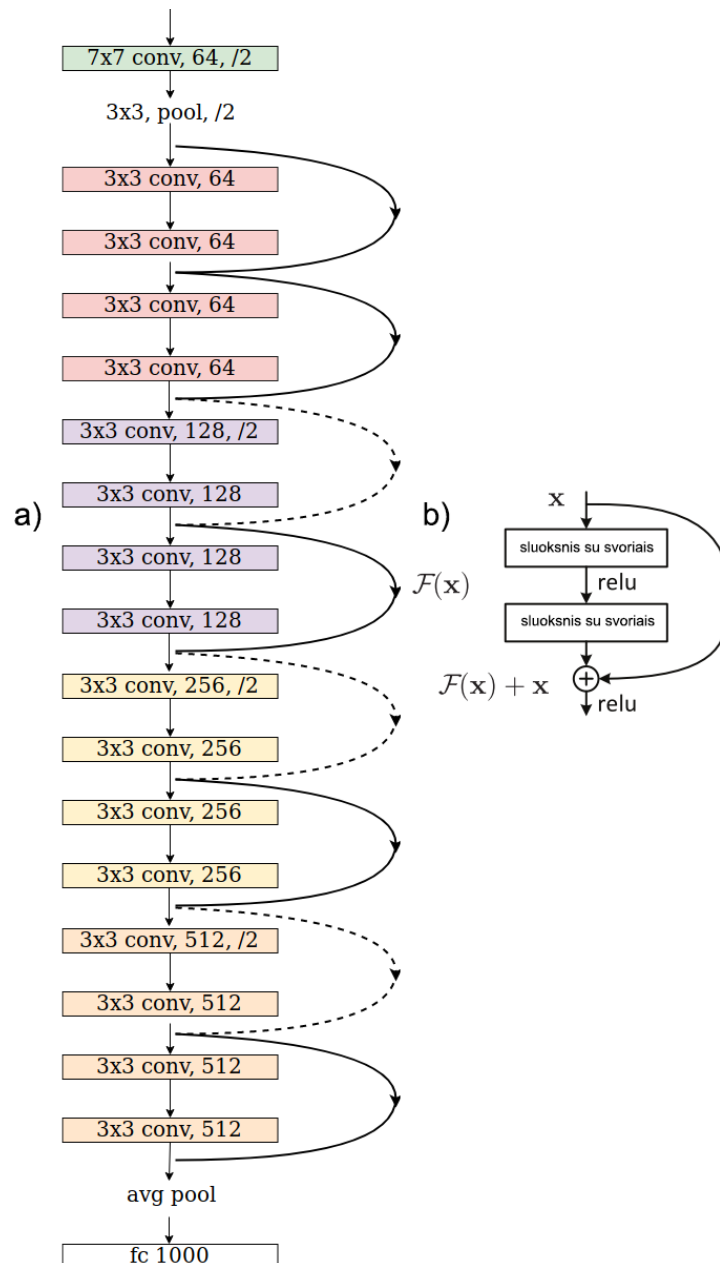
1.1.CenterNet rezultatai

Visi CenterNet eksperimentai buvo atlikti naudojant MS COCO duomenų rinkinį (118 tūkst. mokymo vaizdų, 5 tūkst. validacijos vaizdų ir 20 tūkst. testavimo vaizdų). Įvesties dydis 512 x 512 pikselių. Išvesties dydis visiems tinklams 128 x 128 pikselių. Tinklams apmokyti buvo naudojamas Adam optimizatorius ir L1 tikslo funkcija (2). Tinklams mokytis naudojant mokymąsi su mokytoju (*angl. supervised learning*).

2-oje lentelėje matyti, kad CenterNet tinklas naudojantis Hourglass-104 pasiekė geriausią tikslumą su reliatyviai geru vykdymo greičiu aplenkdamas kitus vienos pakopos (*angl. one-stage*) detektorius. Naudojant šią architektūrą tinklas aplenkė CornerNet [LD18] ir ExtremeNet [ZZK19] (su ta pačia architektūra), kurie irgi objektus identifikuoja kaip taškus. CornerNet objektą aptinka kaip du taškus - viršutinį kairinį tašką ir apatinį dešinįjį tašką, iš kurių galima gauti gaubiantį stačiakampį (*angl. bounding box*). ExtremeNet objektą identifikuoja keturiais taškais, kurie sudaro

⁵ <http://cocodataset.org/>

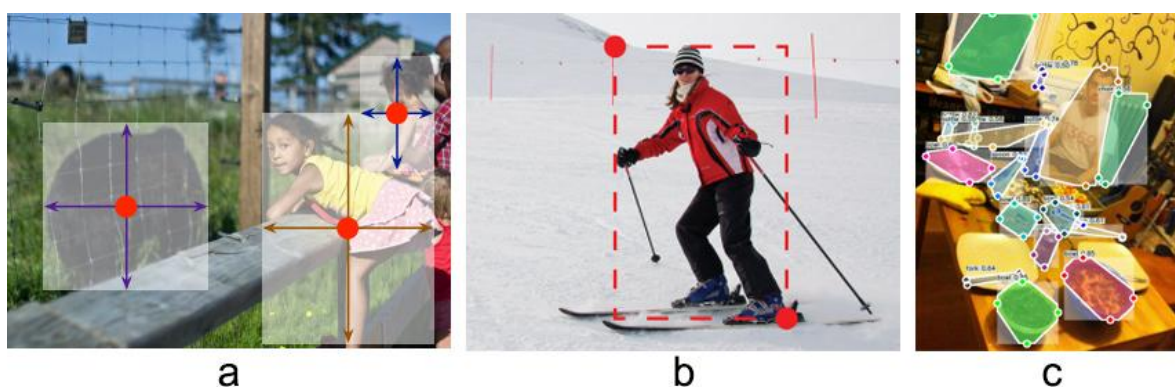
gaubiantį stačiakampį. Paveikslėlyje nr. 7 vaizduojami skirtumai tarp CenterNet, CornerNet ir ExtremeNet objektų aptikimo taškais. DLA-34 architektūra duoda vidutiniškai geriausią rezultatą įvertinus tikslumą ir vykdymo laiką. Pats CenterNet tinklas nėra tiksliausias iš panašių tinklų, tačiau jis tikrai yra greičiausias įvertinus kadro per sekundę greitį (*angl. frames per second (FPS)*). O vykdymo greitis bepiločiame orlaivyje yra svarbus parametras.



6 pav. a) ResNet-18 architektūra b) gradientų “peršokimas” ir sujungimas ResNet tinkle. Šaltinis [HZS+15].

2 lentelė. CenterNet palyginimas su kitais tinklais. Šaltinis [ZWK19].

	Backbone	FPS	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
MaskRCNN [21]	ResNeXt-101	11	39.8	62.3	43.4	22.1	43.2	51.2
Deform-v2 [63]	ResNet-101	-	46.0	67.9	50.8	27.8	49.1	59.5
SNIPER [48]	DPN-98	2.5	46.1	67.0	51.6	29.6	48.9	58.1
PANet [35]	ResNeXt-101	-	47.4	67.2	51.8	30.1	51.7	60.0
TridentNet [31]	ResNet-101-DCN	0.7	48.4	69.7	53.5	31.8	51.3	60.3
YOLOv3 [45]	DarkNet-53	20	33.0	57.9	34.4	18.3	25.4	41.9
RetinaNet [33]	ResNeXt-101-FPN	5.4	40.8	61.1	44.1	24.1	44.2	51.2
RefineDet [59]	ResNet-101	-	36.4 / 41.8	57.5 / 62.9	39.5 / 45.7	16.6 / 25.6	39.9 / 45.1	51.4 / 54.1
CornerNet [30]	Hourglass-104	4.1	40.5 / 42.1	56.5 / 57.8	43.1 / 45.3	19.4 / 20.8	42.7 / 44.8	53.9 / 56.7
ExtremeNet [61]	Hourglass-104	3.1	40.2 / 43.7	55.5 / 60.5	43.2 / 47.0	20.4 / 24.1	43.2 / 46.9	53.1 / 57.6
FSAF [62]	ResNeXt-101	2.7	42.9 / 44.6	63.8 / 65.2	46.3 / 48.6	26.6 / 29.7	46.2 / 47.1	52.7 / 54.6
CenterNet-DLA	DLA-34	28	39.2 / 41.6	57.1 / 60.3	42.8 / 45.1	19.9 / 21.5	43.0 / 43.9	51.4 / 56.0
CenterNet-HG	Hourglass-104	7.8	42.1 / 45.1	61.1 / 63.9	45.9 / 49.3	24.1 / 26.6	45.5 / 47.1	52.8 / 57.7



7 pav. Objektų aptikimas taškais a) CenterNet [ZWK19] b) CornerNet [LD18] c) ExtremeNet [ZZK19].

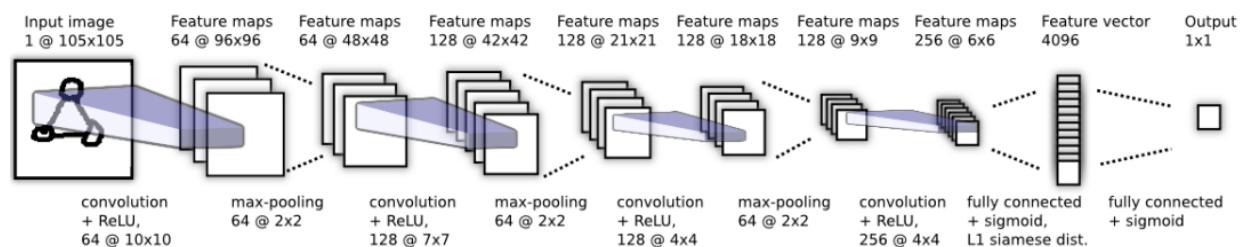
1.2. Vaizdų panašumas

Neuroniniai tinklai puikiai moka atpažinti ir klasifikuoti objektus. Kai klasifikuojamų klasių kiekis yra šimtas ar tūkstantis tinklai puikiai susitvarko su keliama užduotimi, bet jei klasių skaičius būtų milijonai standartiniai klasifikavimo sprendimai sunkiai padėtų. Tipiniai gilieji neuroniniai klasifikavimo tinklai negali susitvarkyti su tokiais uždaviniais, nes klasių kiekis yra per didelis paskutiniam išvedimo sluoksniui (*angl. output*). Todėl reikalingas kitos sprendimas. Tinklo grąžinamas rezultatas turėtų būti unikalūs kiekvienam įvesties vaizdui, tačiau esant panašioms vaizdams, rezultatas turi būti panašus.

Toks rezultatas gali būti požymių vektorių aibė (*angl. embeddings*). Taip, kad daugiamatėje vektorių erdvėje panašūs vaizdai būtų kaimyniniuose regionuose, o nepanašus toliau vienas nuo kito. Paveikslėlio požymių vektoriai tai aibė skaičių, kurie atitinka įvesties paveikslėlį. Požymių vektoriai paprastai naudojami vaizdų panašumui lyginti ar vaizdų klasifikavimui. Bet vien surasti požymių vektorius neužtenka. Reikalinga funkcija juos palyginti. Kadangi vaizdai yra išreikšti vektorių pavidalu, tai tokia funkcija galime vadinti panašumų funkcija, kurią galima

matuoti kokia nors atstumo metrika. Yra daugybė įvairių atstumo metrikų: kosinusas [Pra18], Euklidinis atstumas, p-norma, Pearsono koreliacija, artimiausi kaimynai, (*angl. nearest neighbors*) šablonų dažnumas [BBB+17] (*angl. pattern intensity*). Palyginimui galima naudoti ir įvairias bibliotekas sukurtas didelių komercinių kompanijų - annoy (Spotify)⁶, faiss (Facebook)⁷.

Galima išskirti du pagrindinius giliųjų neuroniniu tinklu architektūrų tipus skirtus vaizdų panašumui nustatyti. Tai siamo tinklai ir tripleto tinklai. Siamo tinklai [KZS15] - arba dar vadinami dvynių tinklai. Tradiciškai tinklai išmoksta atpažinti kokias klasei objektas priklauso, tam reikia daugybės sužymėtų duomenų. Norint pridėti naują klasę, kurią tinklas mokėtų atpažinti, reikia turėti daug vaizdų priskirtų tai klasei ir vėl apmokyti tinklą. Siamo tinklų atveju tinklas nesimoko priskirti objekto konkrečiai klasei, bet bando išmokti panašumo, kiek vaizdas yra panašus į konkrečios klasės vaizdą. Siamo tinklai sudaryti iš dvejų identiškų tinklų, kurie dalinasi tais pačiais tinklo svoriais. Tinklo įvestis yra du vaizdai: konkrečios žinomos klasės paveikslėlis ir paveikslėlis, kuriam siekiama nustatyti ar jis priklauso tai pačiai klasei. Tinklas apskaičiuoja abiem paveikslėliams požymių vektorius, naudodamas L1 (1) tikslo funkciją apskaičiuoją vaizdų panašumą. Paskutiniame sluoksnyje yra sigmoidinė (7) aktyvacijos funkcija, kuri gražina rezultatą tarp 1 ir -1. Rezultatui esant arčiau 1 vaizdai laikomi panašiais, esant arčiau -1 nepanašiais. Paveikslėlyje nr. 8 matoma siamo tinklo schema. Plačiau galima susipažinti su tokiu tinklu Toronto mokslininkų darbe [KZS15].



8 pav. Siamo tinklo architektūra. Šaltinis [KZS15].

Ir tripleto neuroniniai tinklai yra panašūs siamo neuroninius tinklus. Tai trigubi neuroniniai tinklai, kurie dalinasi tais pačiais tinklo svoriais. Tinklų įvestis yra trys paveikslėliai. Toki tinklai naudoja tripleto tikslo funkciją mokymuisi (skyrus 2.6.).

Google mokslininkų sukurtas neuroninis tinklas FaceNet [SKP15] naudojantis tripleto tikslo (3) funkcija davė neįtikėtina gerus rezultatus. Tuo metu egzistavusio tiksliausio veidų

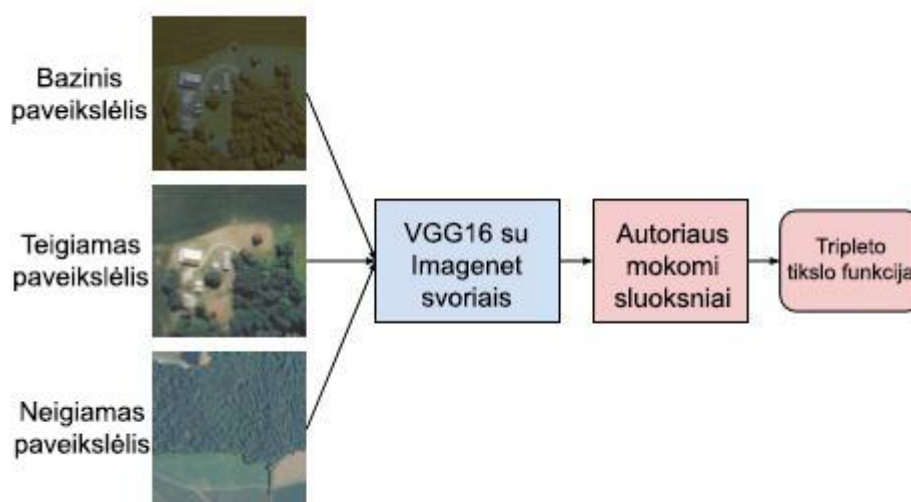
⁶ <https://github.com/spotify/annoy>

⁷ <https://github.com/facebookresearch/faiss>

atpažinimo tinklo rezultatus pagerindamas apie 30% naudojant LFW⁸ (*angl. Labeled Faces in the Wild*) ir YouTube Faces DB⁹ duomenų rinkinius [SKP15].

Panašus darbas, lyginantis ortofotografinių vaizdų panašumą, yra atliktas Vilniaus universiteto mokslininkų [VMR20]. Darbe siūloma tripleto neuroninio tinklo architektūra naudojanti VGG16 neuroninį tinklą ir gale pridėjus mokomą pilnai sujungtą (*angl. fully connected*) sluoksnį. Paveikslėlyje nr. 9 pavaizduota siūloma tinklo architektūra. Mokymui VGG16 sluoksniai yra “užšaldomi”, mokymasis juose nevyksta, mokomi tik paskutiniai pridėti sluoksniai. Darbe taip pat naudojama modifikuota tripleto tikslo funkcija (4). Apie tripleto tikslo funkcijas kalbama skyriuje 2.6. Darbe apžvelgiamos ir įvairios metrikos lyginti vaizdams. Pasak autorių ankstesnio tyrimo [JM19], Pearsono koreliacija yra viena geresnių metrikų matuoti ortofotografinių vaizdų panašumui nustatyti. Tyrimo metu autoriai sugebėjo pasiekti tikslumą didesnę nei 99% tikslumą su geriausia VGG16 konfigūracija, naudojant jų pačių sukurtą duomenų rinkinį.

Dėl gerų rezultatų vertinant ortofotografinių vaizdų panašumą [VMR20] ir geru rezultatų vaizdų panašumo atpažinime naudojant tripleto tikslo funkciją [SKP15], darbe tai pat pasirinkta naudoti tripleto tinklo architektūra.



9 pav. Siūloma neuroninio tinklo architektūra pagrindui naudojat VGG16 tinklą. Šaltinis [VMR20].

1.3. Tripleto tikslo funkcijos

Galima pasakyti, jog neuroninio tinklo mokymas vyksta panašiai, kaip mokomės ir mes. Neuroniniui tinklui pateikiame kokią nors informaciją. Tinklas prognozuoja atsakymą ir mes jam pasakome ar jo prognozės yra teisingos ar ne. Žinodamas klaidas tinklas koreguoja savo prognozių spėjimus. Tai kartojama, kol tinklo prognozuojamo rezultato tikslumas yra tenkinamas. Pasakyti

⁸ <http://vis-www.cs.umass.edu/lfw/>

⁹ <https://www.cs.tau.ac.il/~wolf/ytfaces/>

tinklui, kur jis suklydo prognozuodamas yra labai svarbus momentas, būtent už šią informaciją ir atsakingos tikslo funkcijos. Yra nemažai tikslo funkcijų, galima paminėti keletą iš jų:

MSE (*angl. Mean square error*) - vidutinė kvadratinė paklaida arba dar vadinama L2 tikslo funkcija (1).

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (1)$$

Kaip galima suprasti ir iš pavadinimo, vidutinė kvadratinė paklaida yra kvadratų vidurkio skirtumas tarp tikrosios reikšmės y ir spėjimų rezultatų \hat{y} . n čia yra kiekis visų rezultatų.

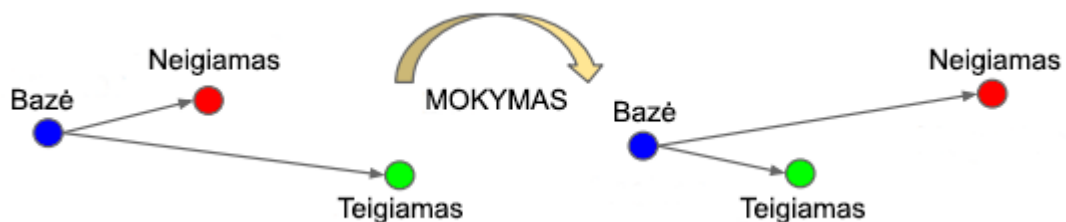
MAE (*angl. Mean Absolute Error*) - vidutinė absoliutinė paklaida arba dar vadinama L1 tikslo funkcija (2).

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (2)$$

MAE priešingai nei MSE nenaudoja kvadratų, todėl yra laikoma labiau atspari duomenų skirtingumams (*angl. robust*).

Nėra neuroninio tinklo tikslo funkcijos, kuri tiktų visiems uždaviniams spręsti. Kiekvienu atveju reikia žiūrėti ko reikalauja mokymo uždavinys. Lyginant vaizdų panašumus galima naudoti tripleto tikslo funkcija. Ši funkcija naudota ir Google mokslininkų darbe apie žmogaus veidų panašumo lyginimus [SKP15], taip pat Vilniaus universiteto mokslininkų darbe apie ortofotografinių vaizdų panašumo metrikas [VMR20].

Tripleto funkcijos esmė - bazinės (*angl. anchor*) įvesties paveikslėlis (požymių vektoriai) yra palyginami su teigiamos (*angl. positive*) ir neigiamos (*angl. negative*) įvesties paveikslėliu. Tikslas yra atstumą tarp bazinės ir teigiamos yra minimizuoti, o tarp bazinės ir neigiamos maksimizuoti. Tai iliustruoja paveikslėlis nr. 10.



10 pav. Atstumas tarp bazinio ir teigiamo (panašaus) ir neigiamo (nepanašaus) vaizdo iki mokymo su tripleto tikslo funkcija ir po jos. Atstumas tarp teigiamo ir bazinio turi mažėti, o tarp bazinio ir neigiamo didėti.

Matematiškai tokia funkcija galima aprašyti taip (3):

$$L(a, p, n) = \sum_{i=1}^{N_{tr}} \max(d(a, p) - d(a, n) + margin, 0) \quad (3)$$

d čia yra kokia nors atstumo metrika, pavyzdžiui Euklidinis atstumas. a, p, n atitinkamai yra požymių vektoriai įvesties vaizdams - baziniam, teigiamam ir neigiamam paveikslėliui. $margin$ yra atotrūkio parametras, kuris reguliuoja atstumą tarp abiejų porų $d(a, p)$ ir $d(a, n)$. $margin$ turi būti daugiau už 0. N_{tr} tripletų skaičius.

Prieš tai aprašytoje tripleto tikslo funkcijoje (3) lyginami du atstumai $d(a, p)$ ir $d(a, n)$. Tačiau dar yra ir trečias atstumas $d(p, n)$, tarp teigiamo ir neigiamo vaizdo. Jį galima irgi panaudoti skaičiuojant tripleto tikslo funkcija. Tam tikrais atvejais kai $d(a, n) < d(p, n)$ galima sukeisti a vietomis su p . Taip bazinis vaizdas tampa teigiamu, o teigiamas baziniu. Tuo užtikrindamas kad “sunkesnis” neigiamas (*angl. hardest negative*) atstumas bus naudojamas atgaliniam tinklo mokymui. Plačiau apie tai galima paskaityti [BRP+16].

Tripleto tikslo funkcijoje (3) yra eilute $\max(f(x), 0)$. Tai reiškia, kad kas kartą esant $f(x) < 0$ informacija bus prarasta. Kai $d(a, p) + margin$ reikšmė yra mažesnė už $d(a, n)$ tinklas negaus jokios naudingos informacijos atgaliniam mokymui. Pavyzdys: jeigu $d(a, p) = 1.2$, $d(a, n) = 2.4$, o $margin = 0.2$, tai $L(a, p, n) = \max(1.2 - 2.4 + 0.2, 0) = \max(-1, 0) = 0$. Tokiais atvejais tinklo apmokymas gali būti labai lėtas. Kad išvengti tokios problemos galima naudoti tripleto tinklo funkcija be nuostolių [Are18]:

$$L(a, p, b) = \sum_{i=1}^N \left[-\ln\left(-\frac{(a-p)^2}{\beta} + 1 + \epsilon\right) - \ln\left(-\frac{N-(a-n)^2}{\beta} + 1 + \epsilon\right) \right] \quad (4)$$

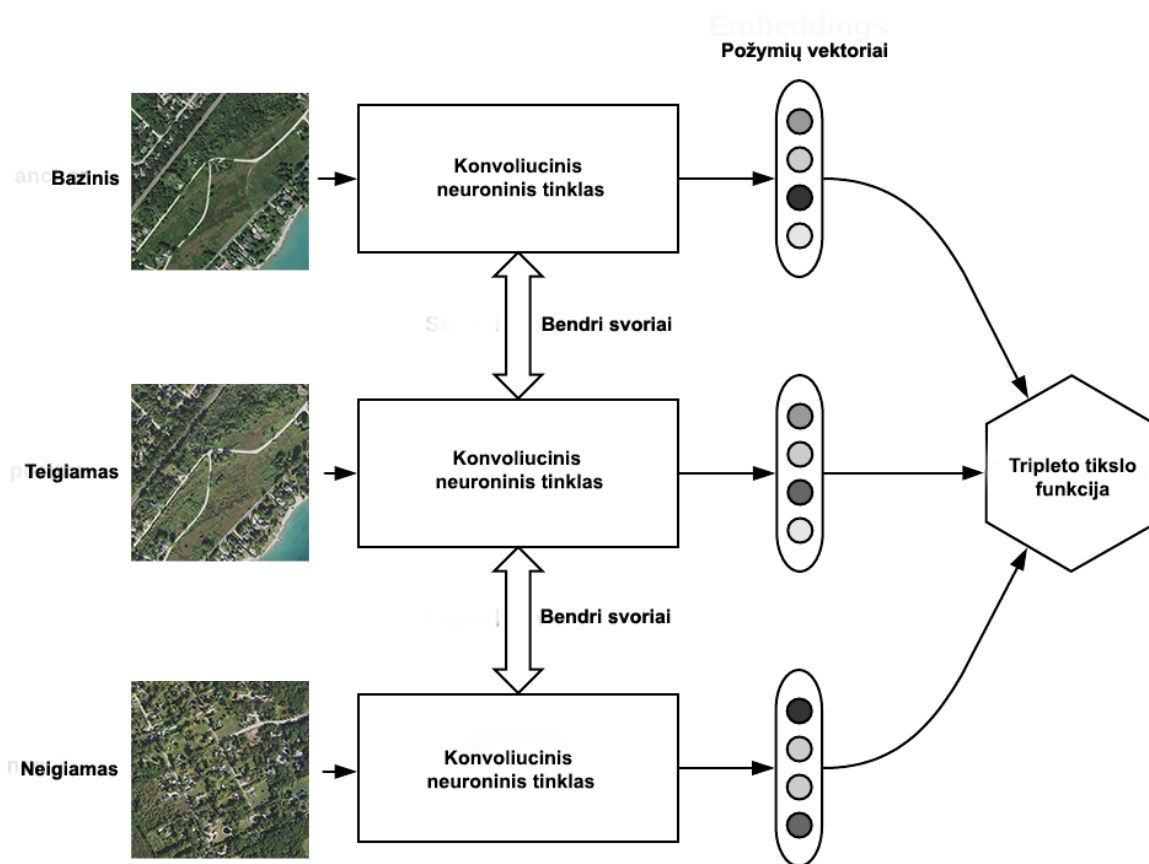
N čia yra dimensijų skaičius, ϵ koks nors mažas skaičius (autorius naudojamas 10^{-6}), β masteliavimo faktorius, autorius siūlo naudoti kokį patį kaip ir dimensijų skaičius. Pastaroji funkcija (4) reikalauja atlikti modifikacijas ir tinklo architektūroje. Paskutinis sluoksnius turi turėti sigmoidinė aktyvacijos funkcija, kad būtų kontroliuojamas požymių vektorių rezultatas tarp 0 ir 1. Šita modifikuota tripleto tikslo funkcija naudojama ir Vilniaus universiteto mokslininkų straipsnyje [VMR20] ji buvo pasiūlyta kaip modifikacija siamo tinklams [KZS15]

Tripleto tikslo funkcijos nėra galutinė riba dimensijų atžvilgiu, kurią galima naudoti mokant gilųjį neuroninį tinklą. Žmonių veidų indentifikavimui mokslininkai išbandė ir tinklus kurie naudoja keturgubą (*angl. quadruplet*) tikslo funkciją [CCZ+17].

1.4. Tripleto tinklo architektūra

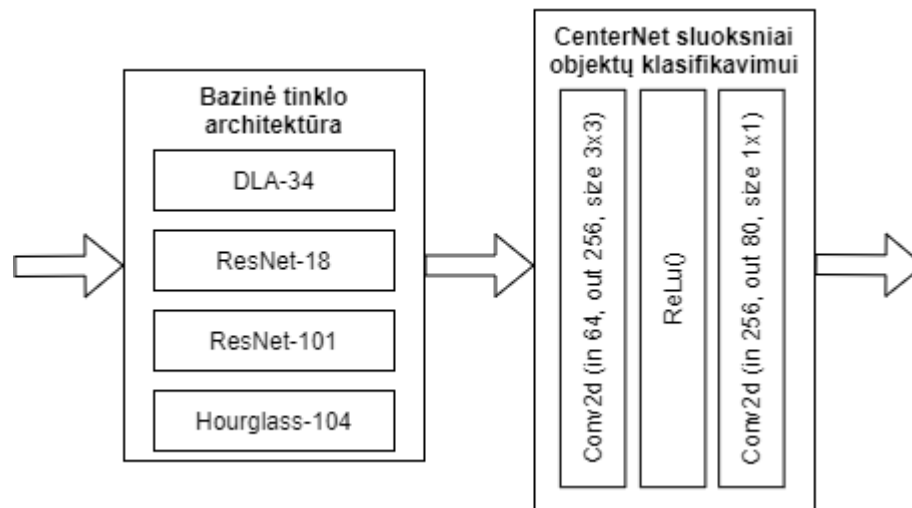
Pagrindinė tripleto neuroninio tinklo esmė yra turėti tris identiškus tinklus, kurie dalinasi bendrais svoriais ir parametrais. Kiekvieno tinklo įvestis yra vienas iš tripleto paveikslėlių (bazinis,

teigiamas, neigiamas). Išvestis - požymių vektorių aibė atitinkanti įvesties paveikslėlį. Apibendrinta tripleto tinklo schema matoma paveikslėlyje nr. 11.



10 pav. Bendra tripleto tinklo architektūra.

Kaip minėta CenterNet [ZWK19] apžvalgos dalyje (skyrius 2.1), tai sistema savyje turinti keletą giliųjų dirbtinių neuroninių tinklų. Iš minėtų keturių tinklų toliau eksperimentui pasirinkti trijų architektūrų tinklai: ResNet-101, ResNet-18 ir DLA-34. Hourglass-104 nebuvo pasirinktas dėl savo architektūros, pagal ją tinklas jau yra dvigubas, jo pritaikymas tripleto tinklui reikalautų kitokių architektūrinių sprendimų, nei naudoti likusiems tinklams. Kiekvienas iš CenterNet tinklų sudarytas iš bazinės tos tinklo architektūros su autoriaus pridėtais 3x3 ir 1x1 konvoliuciniais sluoksniais atskirtais ReLU (6) funkcija, skirtingiems autoriaus uždaviniams spręsti (objektų klasifikavimui, žmonių pozos indentifikavimui, 3D pozicijos nustatymui). Paveikslėlyje nr. 11 parodyta CenterNet tinklo sluoksniai objektų indentifikavimui. Kitiems uždaviniams sluoksniai yra tokie patys, tik skiriasi išvesties (*angl. output*) dydis.



11 pav. CenterNet sluoksniai objektų identifikavimui.

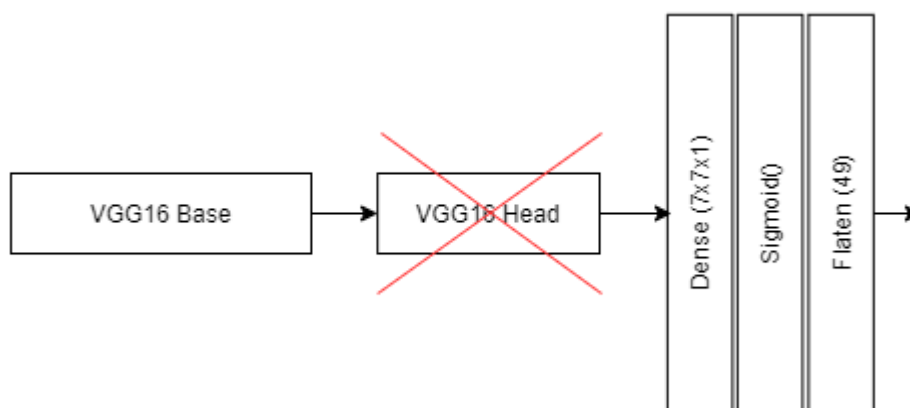
ReLU (*angl. rectified linear unit*) yra aktyvacijos funkcija. Funkcija galima užrašyti taip (6):

$$f(x) = x^+ = \max(0, x) \quad (6)$$

x čia neurono įvestis. ReLU dažnai naudojama kompiuterinės regos ar kalbos atpažinimo (*angl. speech recognition*) uždaviniuose giliuose neuroniniuose tinkluose. Iš aktyvacijos funkcijų galima dar paminėti ir sigmoidinę funkciją (7). Ji yra naudojama minėtuose siamo tinkluose [KZS15] ir ortofotografinių vaizdų atpažinime [VMR20].

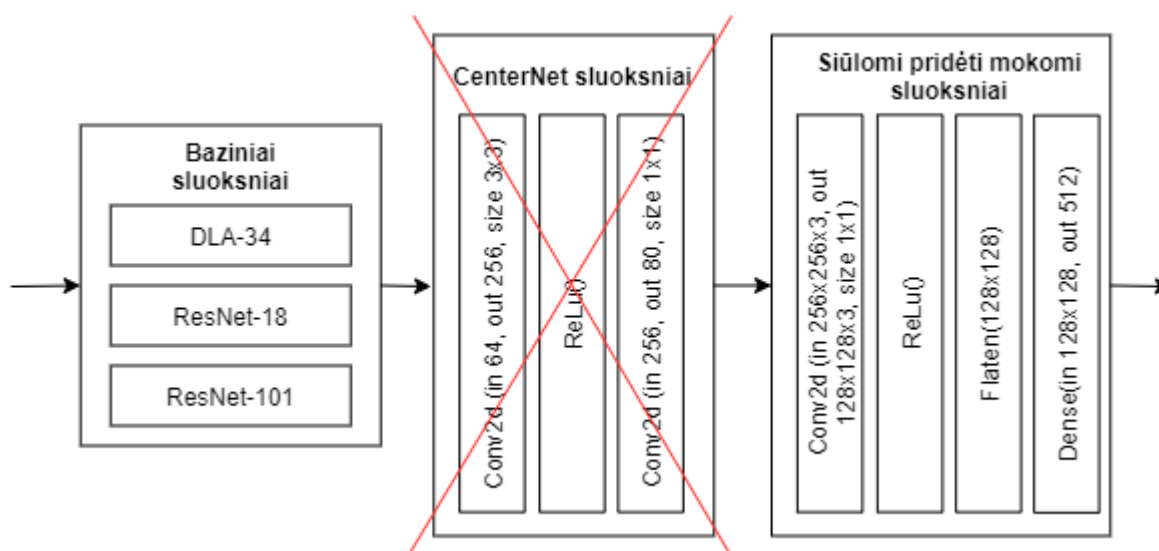
$$f(x) = \frac{1}{1 + e^{-z}} \quad (7)$$

Panašius sluoksnius (kaip naudoja CenterNet, paveikslėlis nr. 11) galima naudoti ir vaizdų vienodumo uždavinyje. Prieš tai minėtame ortofotografinių vaizdų vienodumo darbe [VMR20], autorius pašalina viršutinius (*angl. head*) VGG16 sluoksnius ir prideda du savus mokomus sluoksnius. Pirmasis pilnai sujungtas (*angl. fully connected*) sluoksnis, kurio įvestis priklauso nuo tinklo išvesties tame lygmenyje. Antrasis išlyginamasis sluoksnis, kuris tiesiog sujungia kelių dimensijų masyvą į vienmatį požymių vektorių masyvą tinkama skaičiuoti vaizdų atstumams. Sigmoidinė aktyvacijos funkcija (7) naudojama, kaip to reikalauja naudojama tripleto tikslo funkcija (4) [Are18]. Toks tinklas pavaizduotas 12-tame paveikslėlyje.



12 pav. VGG16 architektūra su papildomais sluoksniais vaizdų panašumui nustatyti [VMR20].

Atsižvelgiant į abi architektūras (CenterNet ir VGG16) galima pritaikyti ir CenterNet tinklą skaičiuoti požymių vektoriams. Pirma pašaliname CenterNet autorių pridėtus sluoksnius, kad liktų tik baziniai tinklų sluoksniai. Tada panašiai kaip CenterNet sluoksniuose pridėkime vieną 1x1 konvoliucinį sluoksnį, kuris atlieka ir dimensijų mažinimo funkcija. Po konvoliucinio sluoksnio seka ReLu aktyvacijos funkcija (6). Sekantis sluoksnis yra išlyginamasis, kuris apjungia kelių dimensijų masyvą į vienmatį, panašiai kaip ir VGG16 tinkle. Ir gale naudojama vienas pilnai sujungtas (*angl. fully connected*) sluoksnis su išvesties dydžiu 512. Tokia išvestis pasirinkta dėl tinklo įvesties paveikslėlių dydžio - 512 pikselių aukščio ir pločio. Siūlomo tinklo architektūra matyti paveikslėlyje nr. 13.



13 pav. Siūloma CenterNet architektūra vaizdų panašumui lyginti.

1.5. Mokymo perdavimas

Tinklo mokymas nuo pradžių sudėtingas ir ilgas procesas reikalaujantis labai didelio kiekio sužymėtų (*angl. labeled*) duomenų. Konvoliuciniam tinklui apmokyti, kad pasiektų norimą

tikslumą gali reikėti dešimčių, šimtų tūkstančių ar net daugiau sužymėtų paveikslėlių. Vien tokios informacijos paruošimas gali atimti labai daug laiko. Todėl tinklų mokymams dažniausiai naudojami vieši duomenų rinkiniai. Vieni populiariausių tokių rinkinių yra:

- MNIST¹⁰ - 70000 ranka rašytų skaičių (nuo 0 iki 9) paveikslėlių duomenų rinkinys,
- MS-COCO¹¹ virš 200000 (sužymėtų) paveikslėlių su realiais objektais (viso per 1.5 milijono objektų paveikslėliuose, 80 objektų kategorijų);
- ImageNet¹² - apie 1.5 milijono paveikslėlių, kiekvienas su informacija apie objektus jame ir juos gaubiančiu stačiakampiu (*angl. bounding box*).

Taip pat galima paminėti Open Images Dataset, CIFAR-10 duomenų rinkinius. Nors minėti duomenų rinkiniai yra labiau bendrinės paskirties, tačiau egzistuoja ir labiau konkrečiai sričiai priklausančių duomenų rinkinių - medicinos¹³ ar agrokultūros¹⁴ tyrimams.

Tačiau net ir turint duomenų rinkinius, tinklo mokymas nuo pradžių yra ilgas uždavinys. Todėl čia gali pasitarnauti žinių perdavimas (*angl. transfer learning*). Mokymo perdavimas yra dažnas reiškinys gilių neuroninių tinklų mokyme. Tai vienas iš mašininio mokymo uždavinių, kaip jau sukauptas žinias galima panaudoti spręsti kitai, bet susijusiai problemai.

Neuroninis tinklas išmoksta atlikti sau priskirtą uždavinį mokymo metu apdorodamas daugybę įvestų duomenų ir mokydamasis iš klaidų. Išmoktos “žinios” yra saugomos to tinklo svoriuose (*angl. weights*). Šitie svoriai “žinios” gali būti perduodami kitam tinklui. Arba paimamas jau apmokytas tinklas ir jis pritaikomas spręsti panašioms uždaviniams, naudojant esamus tinklo svorius. Toks tinklas iškart greičiausiai nebus šimtu procentų tikslus spręsti iškeltam uždaviniui, bet jau nebereikia jo mokyti nuo pradžių, todėl tikėtina, kad jis išmoks naują užduotį spręsti greičiau.

Pavyzdys: jeigu tinklas buvo išmokintas atpažinti automobilius, galima panaudoti jo žinias toliau tinklą mokinti atpažinti sunkvežimiams. Arba jei tinklas išmoko atpažinti dviratį neutralioje aplinkoje, toliau galima mokinti tinklą atpažinti dviratį realioje aplinkoje (ilustracija tokiam mokymui pavaizduota paveikslėlyje nr. 14. Pastarasis sprendimas vadinamas domeno perkėlimu.

¹⁰ <http://yann.lecun.com/exdb/mnist/>

¹¹ <http://cocodataset.org/>

¹² <http://www.image-net.org/>

¹³ <https://www.kaggle.com/xhlulu/recursion-cellular-image-classification-224-jpg>

¹⁴ <https://data.mendeley.com/datasets/4drtyjtfy/1>



14 pav. Skirtingi mokymo domenai tam pačiam objektui.

Bendrai žinių perdavimo ir mokymo proceso eiga galima apibūdinti taip. Imamas tinklo modelis, kuris buvo apmokytas su didžiuoju duomenų rinkiniu. Jis tarnaus, kaip įrankis savybių atpažinimui (*angl. feature extraction*). Paskutiniai tokio tinklo sluoksniai yra pakeičiami savais sluoksniais, kurie bus atsakingi už norimo spręsti uždavinio rezultatą. Baziniai tinklo sluoksniai yra “užšaldomi”, tai yra mokymosi procese jie nedalyvaus, jų svoriai mokymo metu nesikeis. Tai daroma ir minėtame darbe apie vaizdų panašumus [VMR20]. Svorijų šaldymas nėra būtinas dalykas, tačiau reikia atsižvelgti į turimų duomenų mokymui kiekius bei jų skirtingumą. Galimi keturi mokymo scenarijai [GUP17]:

- Pirmas scenarijus kai mokymo duomenų kiekis yra mažas, o duomenų panašumas yra didelis. Tokiu atveju nereikia mokyti viso tinklo. Užtenka tik modifikuoti išvedimo sluoksnius, taip kaip reikalauja užduotis, o visas tinklas tarnaus savybių atpažinimui. Pavyzdžiui turime tinklą kuris buvo apmokytas naudojant ImageNet duomenų rinkinį. O mūsų uždavinys yra tik atpažinti ir atskirti šunis ir kates. Šiuo atveju reikės pakoreguoti paskutinius sluoksnius, kad vietoje 100 klasių išvestų dvi.
- Antras scenarijus kai duomenų yra mažai ir duomenų panašumas yra nedidelis. Tokiu atveju mums reikėtų „užšaldyti“ bazinius sluoksnius ir mokyti tik naujus pridėtus sluoksnius. Toki sluoksniai išmoks atpažinti informacija iš naujų duomenų, o kadangi mokymo duomenų kiekis nėra didelis, tai bus kompensuota bazinių sluoksnių, kurie buvo apmokyti su dideliu duomenų rinkiniu ir dabar „užšaldyti“.
- Trečias scenarijus, kai mokymo duomenų kiekis yra didelis ir panašumas su duomenimis naudojant apmokyti tinklą yra labai mažas. Tokiu atveju geriausia tinklą mokyti nuo pradžių.
- Ketvirtas scenarijus, kai duomenų kiekis yra didelis ir jie yra labai panašūs. Mokymo perdavimas tokiu atveju turėtų duoti geriausius rezultatus. Galime naudoti jau apmokintą modelį ir jį mokyti visą toliau.

Minėti scenarijai yra naudingi tuo atveju, kai jau turime apmokytą tinklą ir norime jį pritaikyti naujam uždaviniui. Bet galimi atvejai kai mokymo informacija yra visai kitame tinkle ar kitokiu pavidalu, nei tinklo svoriai. Tokia problemą nagrinėja straipsnis apie mokymo perdavimą [TSZ+18]. Straipsnio autoriai siūlo mokymo perdavimą kategorizuoti į keturias grupes (lentelė nr. 3).

3 lentelė. Keturios mokymo perdavimo kategorijos [TSZ+18].

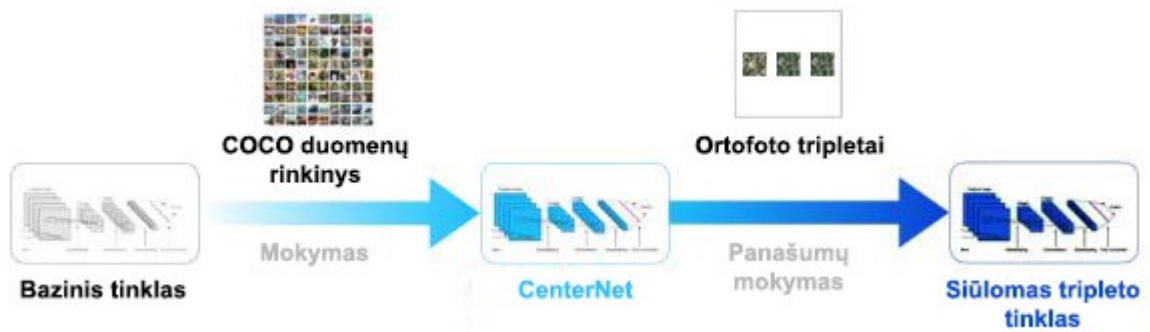
Instancijų (<i>angl. instances-based</i>)	Specifinių savybių ėmimas iš konkretaus domeno papildant norimo domeno duomenis.
Žemėlapiavimo (<i>angl. Mapping-based</i>)	Dvejų duomenų sujungimas iš skirtingų domenų į naują domeną.
Tinklinės (<i>angl. network-based</i>)	Jau esamų apmokytų tinklų naudojimas.
Priešingybių (<i>angl. adversarial-based</i>)	Duomenų ieškojimas priešinguose sistemose (<i>angl. adversarial</i>).

Kito tyrimo metu [RMK05] autoriai priėjo išvada, kad ir esant labai nepanašioms duomenims tarp tų kuriais tinklas buvo mokytas ir norimas mokyti. Mokymo perdavimas dažniausiai padeda, tačiau dėl to gali prailgėti mokymosi laikas.

1.6. CenterNet išankstinis mokymas

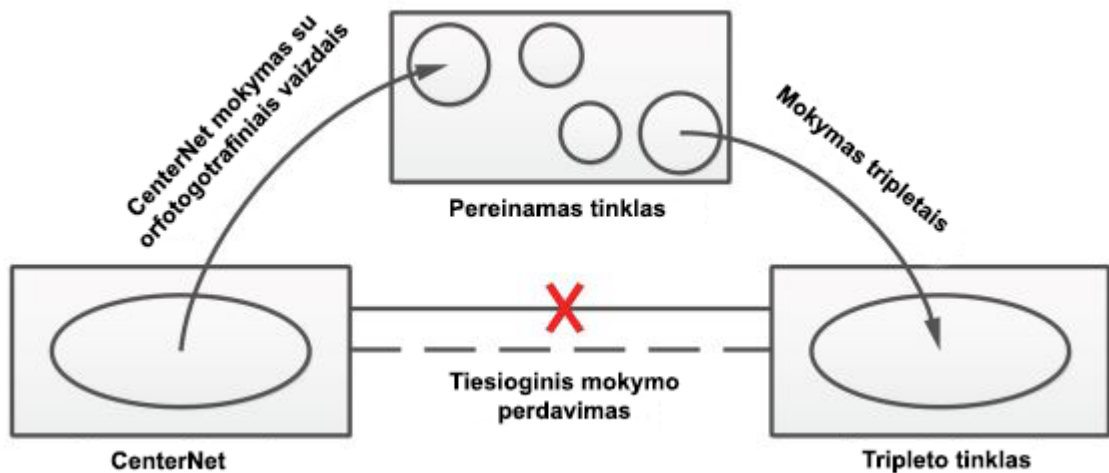
Nėra tinklo modelio gebančio gerai identifikuoti objektus ortofotografiniuose vaizduose, kuris buvo apmokytas nuo pradžių. Pagrindinės priežastys, kad nėra pakankamai didelio ir plataus duomenų rinkinio tam ir tai yra ganėtinai siaura tyrimų sritis. Todėl šiame darbe atliekamas tyrimas, kurio metu bandoma nustatyti, kaip pradiniai svoriai gali įtakoti neuroninių tinklų mokymąsi vaizdų panašumų nustatymui.

Bendra mokymo perdavimo schemą siūlomui ortofotografinių vaizdų atpažinimo tinklui pavaizduota paveikslėlyje nr. 15. Prie bazinių tinklų pridedami sluoksniai apmokomi CenterNet objektų detektoriumi, naudojant COCO duomenų rinkinį. Taip gaunamas apmokytas CenterNet tinklas. CenterNet tinklą modifikuojame, „užšaldydami“ bazinius sluoksnius ir mokome su ortofotografiniais tripletais, vaizdų vienodumui atpažinimui. Taip gaunamas tinklas gebantis įvertinti ortofotografinių vaizdų panašumą.



15 paveikslėlis. Tripleto tinklo mokymo perdavimas.

Yra dar vienas būdas, kurį galima išbandyti perkeliant mokymąsi, tai pereinamasis mokymosi perdavimas (*angl. transitive transfer learning*) [TSZ+15]. Pagrindinė idėja, kad prieš perduodant mokymo konkrečiai užduočiai, šiuo atveju ortofotografinių vaizdų panašumui, pirminis tinklas (CenterNet) mokomas su panašiais duomenimis atlikti savo uždaviniui (objektų atpažinimui), kokias būtų mokomas galutinis tinklas. Taip tikintis, jog papildomo mokymo metu tinklas susidarys kažkokią informaciją apie būsimo užduoties domeną, kas turėtų duoti geresnius rezultatus. Tokio mokymo perdavimo schema pavaizduota paveikslėlyje nr. 16.



16 pav. Pereinamasis mokymo perdavimas.

1.7. Konvolucijos



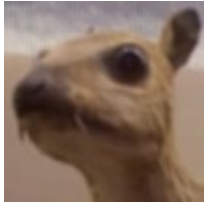
Matematikoje konvolucija yra matematinis operatorius, kuris kaip argumentus paima dvi funkcijas „f“ ir „g“ ir grąžina trečią, kuri, tam tikra prasme, parodo "f" ir "g" persidengimo kiekį. f ir g funkcijų konvolucija žymima $f * g$. Ji apibrėžiama, kaip funkcijų sandaugos integralas, po to, kai viena jų buvo paversta ir padauginta iš -1. Taigi, konvolucija yra integralinės transformacijos rūšis:

$$(f * g)(t) = \int_a^b f(\tau)g(t - \tau)d\tau \quad (8)$$

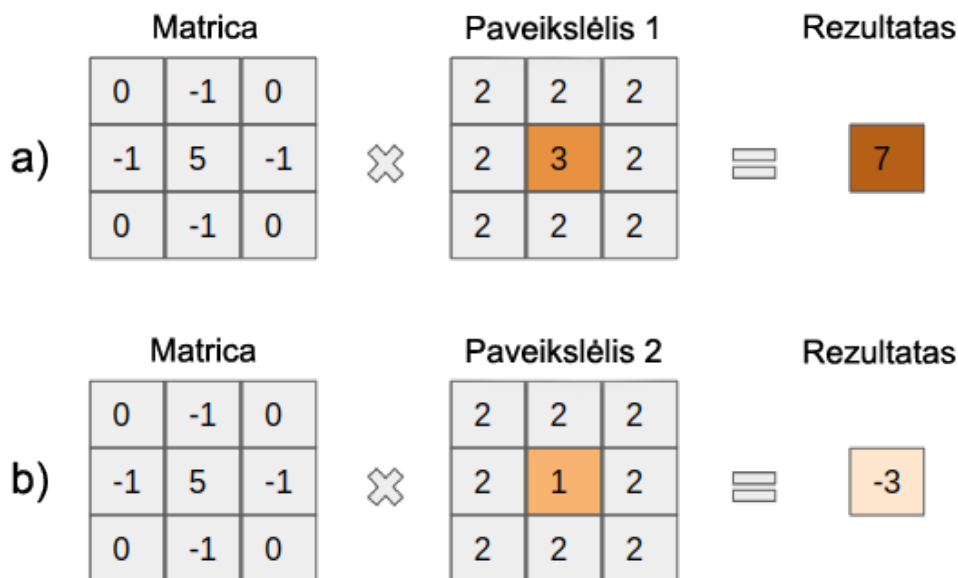
Integravimo rėžiai priklauso nuo funkcijų apibrėžimo srities. Dažniausiai $a = -\infty$ ir $b = +\infty$. Integravimo baigtiniame intervale, f ir g dažnai išplečiamos periodiškai abejomis kryptimis, kad reiškinyje $g(t - \tau)$ argumentas g visada priklausytų apibrėžimo sričiai. Toks periodinis sričių panaudojimas kartais vadinamas cikline arba periodine konvoliucija. Kai apibrėžimo sritis išplečiama, imant funkcijos reikšmes naujuose taškuose lygius 0, konvoliucija vadinama tiesine [SMI97].

Vaizdų apdorojime branduolys (*angl. kernel*) (arba konvoliucinė matrica) yra maža matrica, kuri naudojama išgauti įvairioms paveikslėlio savybės. Pavyzdžiui sulieti (*angl. blur*), paaštrinti (*angl. sharpen*), aptikti kraštams (*angl. edge detection*) bei kitoms [KIP00]. Tai atliekama atlikus konvoliucija tarp paveikslėlio ir minimos matricos. Lentelėje nr. 4 pateikiama keletas konvoliucinių matricų ir kokių rezultatą jos duotų. Vaizdų konvoliucija yra labai panašu į matricų daugyba, skirtumas tas kad daugyba atliekama tik su konkrečiu paveikslėlio pikseliu ir tais jo kaimynais, kokio dydžio yra konvoliucinė matrica. Pavyzdžiui norint paaštrinti paveikslėlį reikėtų naudoti matrica iš antros lentelės nr. 4 eilutės.

4 lentelė. Konvoliucijų su vaizdu pavydžiai. Šaltinis [KIP00].

Operacija	Konvoliucinė matrica	Vaizdo rezultatas
Kraščių aptikimas	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
Savybių ryškinimas	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Suliejimas	$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	

Pavyzdys (paveikslėlis nr. 17). Tarkime yra du paveikslėliai: 1 ir 2. Pirmam paveikslėliui atlikus konvoliucija centrinio pikselio vertė padidėjo iki 7. Tai gauname atlikę veiksmus: $3 \cdot 5 + 2 \cdot (-1) + 2 \cdot (-1) + 2 \cdot (-1) + 2 \cdot (-1) = 7$ (pav. nr. 17 eilutė a). Antram paveikslėliui atlikę konvoliucija gauname $1 \cdot 5 + 2 \cdot (-1) + 2 \cdot (-1) + 2 \cdot (-1) + 2 \cdot (-1) = -3$ (pav. nr. 17 eilutė b). Vienu atveju centrinio pikselio vertė padidėjo iki 7, kitų sumažėjo iki -3. Tai pirmame paveikslėlyje padarytų savybės ryškesnės.

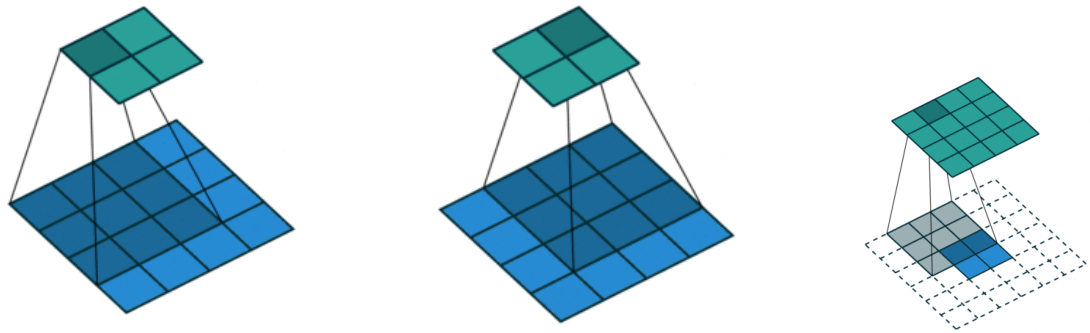


17 pav. Konvoliucijos pavyzdys.

Galima išskirti tris pagrindinius konvoliucijos parametrus:

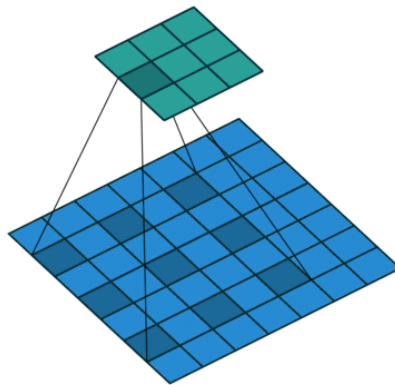
- branduolio dydis (*angl. kernel size*) - tai konvoliucinės matricos dimensijos;
- žingsnis (*angl. stride*) - nusako koku atstumu bus perstumta matrica kitam skaičiavimui;
- paraštės (*angl. padding*) - nusako kaip konvoliucija elgsis prie paveikslėlio kraštų. Priešingai nei kiti parametrai keičiant pastarąjį galima padidinti paveikslėlio dimensijas (*angl. upsampling*). Dėl šios savybės konvoliucijos yra dažnai naudojamos moderniuose neuroniniuose tinkluose.

Minėti parametrai pavaizduoti paveikslėlyje nr. 18. Žalsvai vaizduojamas kvadratas yra konvoliucinė matrica, mėlynai paveikslėlis. Pirmų dviejų vaizdų atveju matricos branduolio dydis yra 2×2 , dešiniausio vaizdo 4×4 . Žingsnis - 1, tai matoma iš kairės ir centrinio vaizdo - per kiek pasislinko matrica kitam skaičiavimui. Dešinys vaizdas parodo kaip atrodytų paraštės su dydžiu 2.



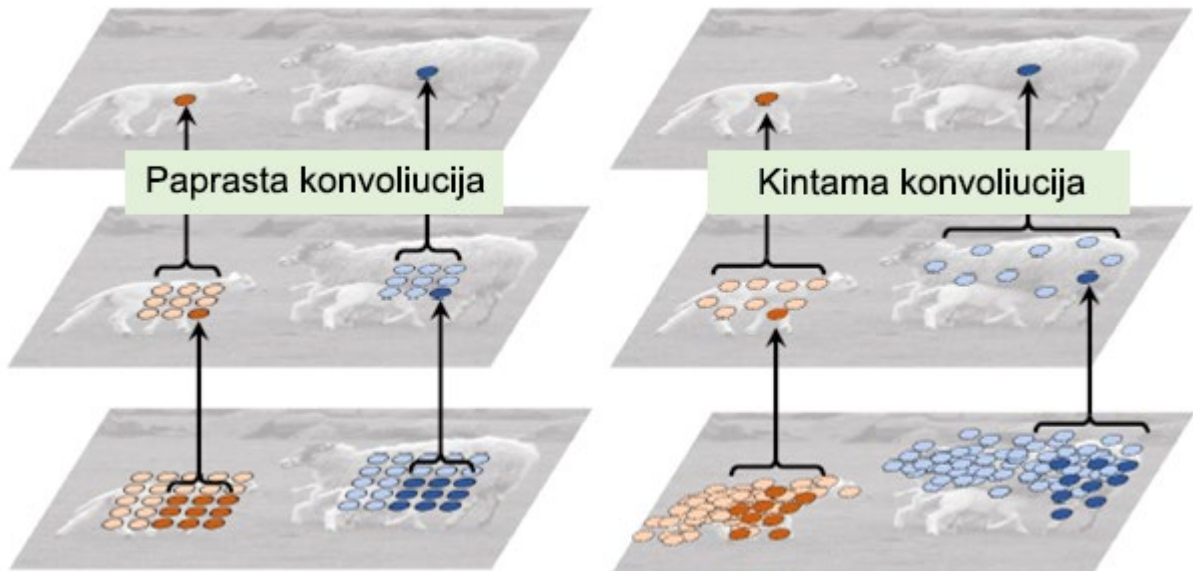
18 pav. Konvoliucijos parametrai.

Kaip matoma iš aukščiau pateiktų pavyzdžių, konvoliucijos atliekamos tik su tais kaimyniniais paveikslėlio pikseliais, kokio dydžio ir yra konvoliucinė matrica, bet nebūtinai turi būti tik taip. Prinstono universiteto ir Intel laboratorijos mokslininkai pasiūlė naudoti išplėstines (*angl. dilated*) konvoliucijas [YK16]. Paveikslėlyje nr. 19 pateiktas išplėstinės konvoliucijos pavyzdys.



19 pav. Išplėstinės konvoliucijos pavyzdys.

Iki šiol minėtos konvoliucijos naudoja kvadrato formos duomenų nuskaitymą iš vaizdo. Tačiau galimos ir kitokios formos. Kintamos (*angl. deformable*) konvoliucijos [DQX+17], pastarosios yra naudojamos ir CenterNet tinkle. Kintama konvoliucija sudaryta iš dviejų matricių. Viena matrica mokinasi poslinkius, antra yra tiesiog matematinė matrica su kuria atliekama konvoliucija. Geriausia kintamas konvoliucijas suprasti iš paveikslėlio nr. 20. Tokios konvoliucijos yra labai lanksčios savybių suradime įvairiems objektams, nes gali išmokti to objekto formą.



20 pav. Paprastos konvoliucijos ir kintamos palyginimas. Šaltinis [DQX+17].

1.8. Metrikos

Turint neuroninį tinklą, kuris generuoja požymių vektorius vaizdui reikalingos ir metrikos galinčios pamatuoti, kaip tiksliai jis tą atlieka. Šiam tyrimui pasirinktos trys metrikos tam atlikti: tikslumas (*angl. accuracy*), vidutinis tikslumas (*angl. average precision*) ir Pearsono koreliacija (*angl. Pearson correlation*).

1.8.1. Tikslumas

Tikslumui (*angl. accuracy*) matuoti reikalinga kokia nors atstumo metrika d , pagal kuria bazinis paveikslėlis a yra arčiau teigiamo paveikslėlio p ir toliau nuo neigiamo n .

$$d(a, p) < d(a, n) \quad (9)$$

Jeigu rezultatas tenkina tokią sąlygą (9) laikoma, jog tinklas teisingai įvertino tripletą. Taip galime paskaičiuoti ir vidurkį visam duomenų rinkiniui N . Labai retas variantas, kad atstumai tarp abiejų būtų lygus, eksperimento metu nebuvo nei vieno tokio atvejo. Todėl jei taip nutiktų laikysime, jog sąlyga nėra tenkinama. Todėl bendrą tikslumą tinklui galime apskaičiuoti:

$$tikslumas = \sum_{i=1}^N \frac{d(a,p) < d(a,n)}{N} \quad (10)$$

N - yra bendras tripletų skaičius. Eksperimente bus naudojamas Euklidinis atstumas, kaip atstumo metrika, bet galima naudoti ir bet kurią kitą atstumo metriką. 1 yra laikomas geriausiu įmanomu rezultatu, 0 prasčiausiu.

1.8.2. Vidutinis tikslumas

Tikslumo (*angl. accuracy*) metrika įvertina atstumus tarp dviejų vaizdų porų a, p ir a, n . Norint turėti geresnį vaizdą, kuris įvertintų konkretaus vaizdo p panašumą visoje sistemoje galima naudoti tikslumo¹⁵ (*angl. precision*) (11) ir atmetimo (*angl. recall*) (12) metrikas.

Tikslumas¹⁵ matuojamas visai sistemai IR . Jei tiksliau, tai yra dalis visų gražintų rezultatų, kurie yra laikomi aktualiais (teisingais). Tegu visa tikrinamų vaizdų aibė būna Q . R yra visi teigiami dokumentai Q aibėje. Sistema IR apdoroja visą aibę Q ir gražina aibę A . Tegu $|R_a|$ būna dokumentų skaičius, kurie egzistuoja R ir A . $|R|$ yra rezultatų skaičius R , $|A|$ yra rezultatų skaičius A . Taip galima apskaičiuoti tikslumą¹⁵ [EY09b]:

$$P = |R_a|/|A| \quad (11)$$

Atmetimas (*angl. recall*) taip pat skaičiuojamas visai sistemai IR . Tai yra dalis iš visų aktualių rezultatų, kuriuos sistema gražina. Laikykime, kad visa sistema yra tokia pati kaip ir tikslumo¹⁵ metrikoje. Tik dabar $|R \cap A|$ yra skaičius dokumentų, kurie priklauso R ir A . Taip galima apskaičiuoti atmetimą (*angl. recall*) [EY09c]:

$$R = |R \cap A|/|R| \quad (12)$$

Žinant tikslumo¹⁵ ir atmetimo metrikas galima apskaičiuoti ir vidutinį tikslumą (*angl. average precision*). Vidutinis tikslumas tai yra kombinacija tikslumo¹⁵ ir atmetimo metrikų surūšiuotai r rezultatų aibei. Tai kiekvieno tikslumo¹⁵ rezultatas, kai randamas teigiamas rezultatas. Tyrimo atveju teigiamas rezultatas yra distancija tarp a_i ir p_i . Vidutinio tikslumo formulė galima užrašyti taip [EY09a]:

$$\text{vidutinis tikslumas} = \frac{\sum_r P@r}{R} \quad (13)$$

¹⁵ Kadangi nėra skirtingo tikslaus vertimo *angl. accuracy* ir *angl. precision* metrikoms. *Precision* bus verčiama - tikslumas¹⁵.

@ čia žymi ribą. Vidutinis tikslumas parodo kiek panašus yra teigiamas paveikslėlis p į bazinį a , lyginant su visais testuojamais teigiamais ir neigiamais vaizdais. Jeigu atstumas tarp $d(a_i, p_i)$ yra pats mažiausias lyginant su visais atstumais tarp $d(a_i, p_x)$ ir $d(a_i, n_x)$, kur p_x yra kiekvienas iš esančių teigiamų vaizdų, o n_x kiekvienas iš esančių neigiamų vaizdų. Toks poros vidutinis tikslumas bus 1, atitinkamai esant kitiems mažesniems atstumam tarp kitų porų šitas skaičius mažės. 1 yra laikomas geriausiu įmanomu rezultatu, 0 prasčiausiu. Eksperimente bus naudojamas Euklidinis atstumas, kaip atstumo metrika, bet galima naudoti ir bet kurią kitą atstumo metriką.

1.8.3. Pearsono koreliacija

[VMR20] Darbe minima, kad Pearsono koreliacija yra labiausiai tinkama metrika skaičiuoti panašumus ortofotografiniuose vaizduose. Pearsono koreliacijos formulė:

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}} \quad (14)$$

Tikslumas apskaičiuotas laikant, kad teigiamas rezultatas yra:

$$pearson(a, p) > pearson(a, n) \quad (15)$$

Turint kiekvienos poros $((a, p), (a, n))$ koreliacijos reikšmes. Galima rasti koreliacijos koeficientą kuriam esant vaizdų pora laikoma panašia ir taip pamatuoti tinklo modelio tikslumą. Testuojamų teigiamų vaizdų tikslumą pagal koeficiento ribą galima užrašyti taip (16):

$$ppa = \sum_{i=1}^N \frac{pearson(a, p) > kk}{N} \quad (16)$$

Čia kk yra pasirinktas koreliacijos koeficientas. Viso teigiamos $pearson(a, p)$ reikšmės, kurios yra didesnės už jį yra laikomos teisingai įvertintomis ir paskaičiuojamas jų procentas visai porų aibei N .

Iš to gauname kad egzistuota ir neigiamų reikšmių kiekis, kuris tenkina panašumo ribą kk . Jas galima apskaičiuoti taip pat kaip testuojamų teigiamų vaizdų (16), tik naudojama kita Pearsono koreliacijos pora (17):

$$pna = \sum_{i=1}^N \frac{pearson(a, n) \geq kk}{N} \quad (17)$$

2. Eksperimentai

Visi eksperimentai buvo atlikti naudojant Pytorch 1.3.1 karkasą, Python 3.7, Nvidia Geforce GTX 1060 Max-Q, Intel i7-8750H, 16GB RAM, Linux Ubuntu 14.08.

Eksperimente naudojami trys modifikuoti CenterNet tinklo modeliai: DLA-34, ResNet-18 ir ResNet-101, kurie aprašyti CenterNet tinklo apžvalgos dalyje (skyrius 2.2). Modeliai modifikuoti identišškai, tai yra pašalinus CenterNet tinklo sluoksnius ir pridėjus savus sluoksnius, kurie aprašyti tinklo architektūros dalyje (skyrius 2.7). Mokomi yra tik pridėti sluoksniai, bazinių sluoksnių mokymas yra „užšaldytas“.

Galima išskirti eksperimentą į dvi dalis. Pirmoje dalyje nagrinėjama, kokią įtaką vaizdų atpažinimui daro mokymo perdavimas (skyrius 2.8). Antroje dalyje lyginamos tripleto tikslo funkcijos (skyrius 2.6) ir kaip jos įtakoja tinklo rezultatus. Abu eksperimentai lyginami metrikomis aprašytomis skyriuje 2.11.

Eksperimentų metu tripleto tikslo formulėje (3) ir jos variante su pakeitimais naudojamas *margin* parametras yra 0.4.

2.1. Duomenų rinkinys

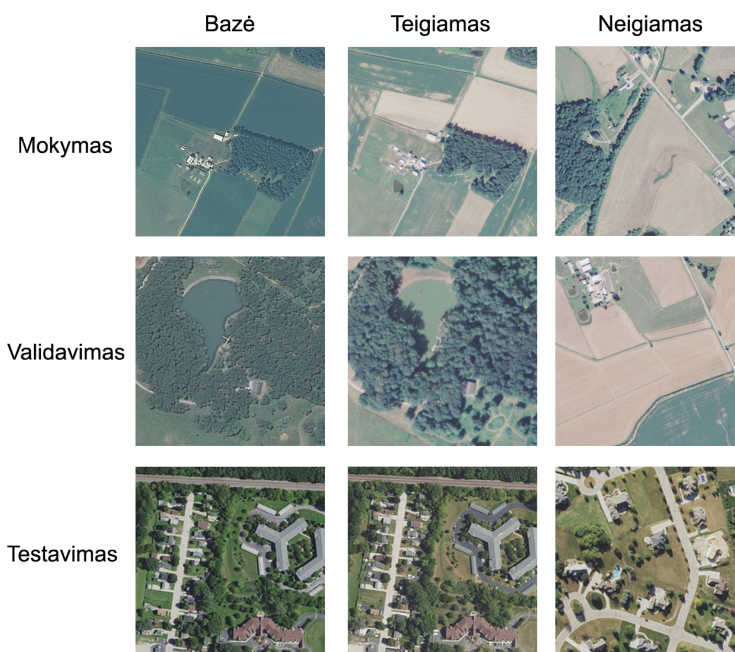
Žemėlapiai tripleto vaizdams generuoti paimti iš NAIP¹⁶ (*angl. National Agriculture Imagery Program*) ir sukarpyti naudojant atsitiktinės sekos (*angl. random*) algoritmą. Pavyzdys kaip atrodo ortofotografinių vaizdų tripletas pateiktas paveikslėlyje nr. 21. Mokymui sugeneruota 18000 nuotraukų (arba 6000 tripletų), 1800 (600 tripletų) iš jų buvo naudojami validacijai, testavimui sugeneruota 600 paveikslėliu (200 tripletų). Mokymo vaizdai baziniam ir teigiamam paveikslėliui, sugeneruoti iš dviejų žemėlapių, tarp kurių yra poros metų skirtumas. Neigiamas paveikslėlis paimtas iš atsitiktinės vietos teigiamo paveikslėlio žemėlapiu. Testavimo aibė generuota tuo pačiu principu, tik iš naujų žemėlapių, kurie nebuvo naudoti mokymosi vaizdams gauti. Paveikslėliai yra 512 x 512 pikselių dydžio be kameros vaizdo iškreipimo (*angl. camera distortion*).

2.2. Tinklo mokymas

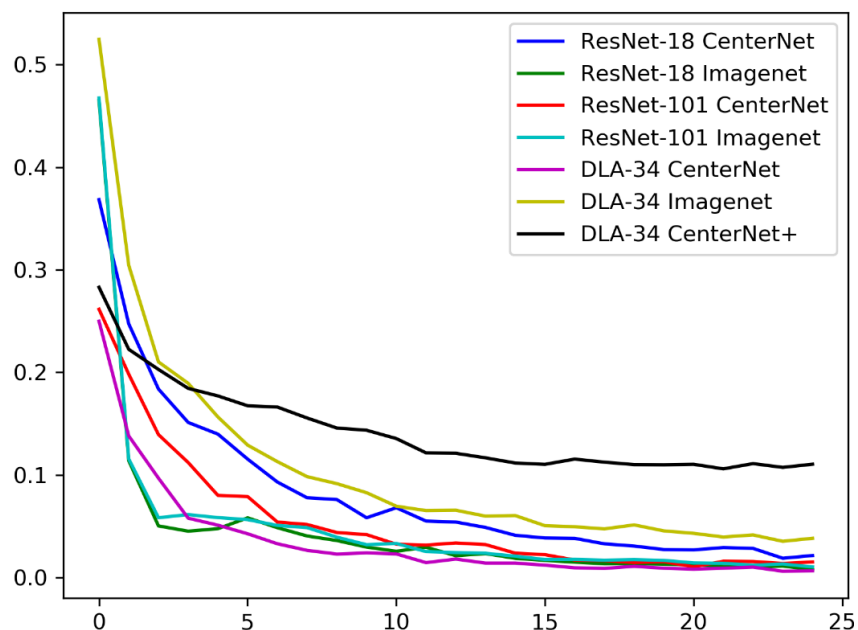
Visų eksperimentų tinklo modeliai buvo mokinami tais pačiais duomenimis. Naudojant Adam optimizatorių ir atitinkamas tripleto tikslo funkcijas, kurios yra testuojamos. Mokymo paklaidos grafikas matyti paveikslėlyje nr. 22. Paveikslėlyje nr. 23 pavaizduota tinklo modelių validacijos

¹⁶ <https://www.fsa.usda.gov/programs-and-services/aerial-photography/index>

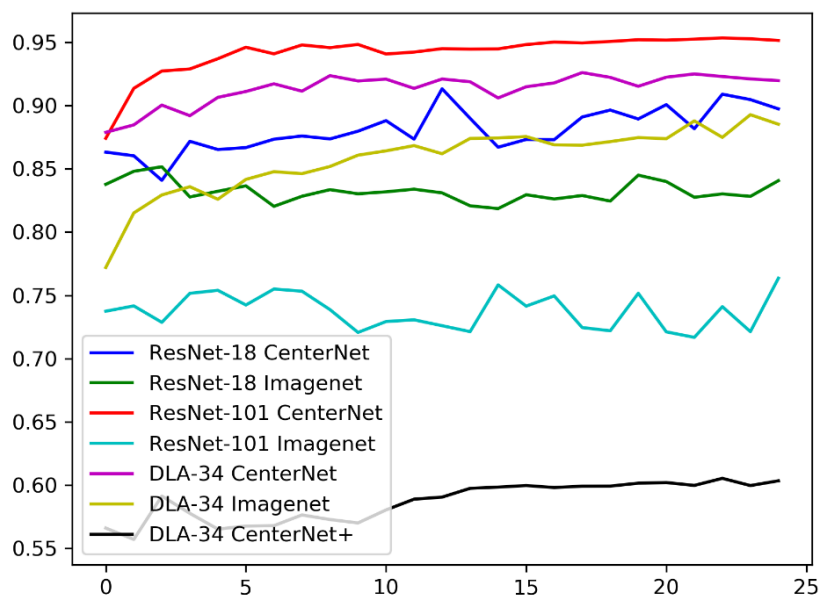
vidutinis tikslumas (13). Šiuo atveju validacija nebuvo naudojama tinklo stabdymui ar parametru keitimui. Grafike matyti, kad dalies tinklo modelių vidutinis tikslumas didėja iki 10 epochos, toliau prasideda vidutinio tikslumo „šokinėjimai“ ir tinklai nebesugeba efektyviai mokytis. Remiantis tokiais duomenimis, eksperimentuose bus rodomi rezultatai iki 26 tinklo mokymo epochos, dažniausiai tinklo modeliai geriausius rezultatus pasiekia apie 7-20 epochoje, todėl pasirinktas toks intervalas.



21 pav. Duomenų rinkinio pavyzdys mokymui, validacijai ir testavimui.



22 pav. Tinklo modelių mokymo paklaidos.



23 pav. Tinklo modelių validacijos tikslumas.

2.3. Eksperimentai su išankstiniu mokymu

2.8 skyriuje yra apžvelgiama mokymosi perdavimo (angl. transfer learning) ypatumai, remiantis tuo parinkta trys skirtingi svoriai tinklams:

- CenterNet svoriai - tai tinklo autorių duoti CenterNet tinklo svoriai, kurie gauti tinklą apmokius su COCO duomenų rinkiniu ir CenterNet objektų detektoriumi,
- ImageNet svoriai - tai svoriai, kurie gauti bazinius tinklus apmokant ImageNet duomenų rinkiniu¹⁷.
- CenterNet+ - tai svoriai, kurie gauti apmokius CenterNet tinklą ortofotografiniais vaizdais aptikti žemėlapių vaizdo lokacijas. Taip gaunant tarpinio mokymo svorius (skyrius 2.8). Šitas eksperimentas atliktas tik su DLA-34 architektūra ir plačiau aprašytas sekancioje dalyje (skyrius 2.3.1.).

Tikimasi pamatyti, kaip skirtingi pradiniai tinklo svoriai įtakoja tinklo mokymąsi ortofotografinių vaizdų panašumui atpažinti. Ar tinklai sugebės pasiekti tokį patį tikslumą ir kaip greitai jie tai padarys. Eksperimento matrica matyti lentelėje nr. 5.

5 lentelė. Mokymosi perdavimo eksperimentų matrica.

Tinklas \ Svoriai	CenterNet	ImageNet	CenterNet+
ResNet-18	ResNet-18 CenterNet	ResNet-18 ImageNet	
ResNet-101	ResNet-101 CenterNet	ResNet-101 ImageNet	
DLA-34	DLA-34 CenterNet	DLA-34 ImageNet	DLA-34 CenterNet+

¹⁷ <http://dl.yf.io/dla/models/imagenet/>

2.3.1. Tarpinis mokymas

Sugeneruoti tarpinius tinklo svorius (CenterNet+) CenterNet DLA-34 architektūros tinklas buvo apmokytas ortofotografiniais vaizdais. Vaizdai gauti tokiu pat būdu kaip ir tiriamojo duomenų rinkinio (skyrius 3.1), tik vietoje tripletų imamas vienas paveikslėlis ir jį atitinkanti koordinatė žemėlapyje. Taip tinklui pateikta mokymo metu 60000 žemėlapių vaizdų su koordinatėmis. Darant prielaidą, kad tinklas sugebės išmokti atpažinti lokaciją ir grąžins tos lokacijos koordinatas. Kadangi mokymo duomenų rinkinys nėra didelis, daroma prielaida, kad tinklas išmoks žemėlapi „mintinai“, tačiau žemiausi sluoksniai turėtų išmokti savybes, kurios yra artimos ortofotografiniams vaizdams. Taip pagreitinant mokymą vaizdų panašumui atpažinti mokant tripleto tinklais. Geriausios epochos tokio mokymo tinklo svoriai (CenterNet+) yra naudojami eksperimente. Prognozuojama, kad tinklas naudojantis šituos svorius pasieks geriausius rezultatus greičiausiai.

2.3.2. Rezultatai

Lentelėje nr. 6 pateikiamas visų eksperimento bandymų (lentelė nr. 5) tikslumas (10). Žalsva spalva žymimas geriausias rezultatas, pažymimi visi modeliai ir epochos, kuriems pavyko pasiekti geriausią rezultatą. Rausva spalva pažymima geriausias konkretaus modelio rezultatas, jeigu jis nėra geriausiai tos lentelės rezultatas.

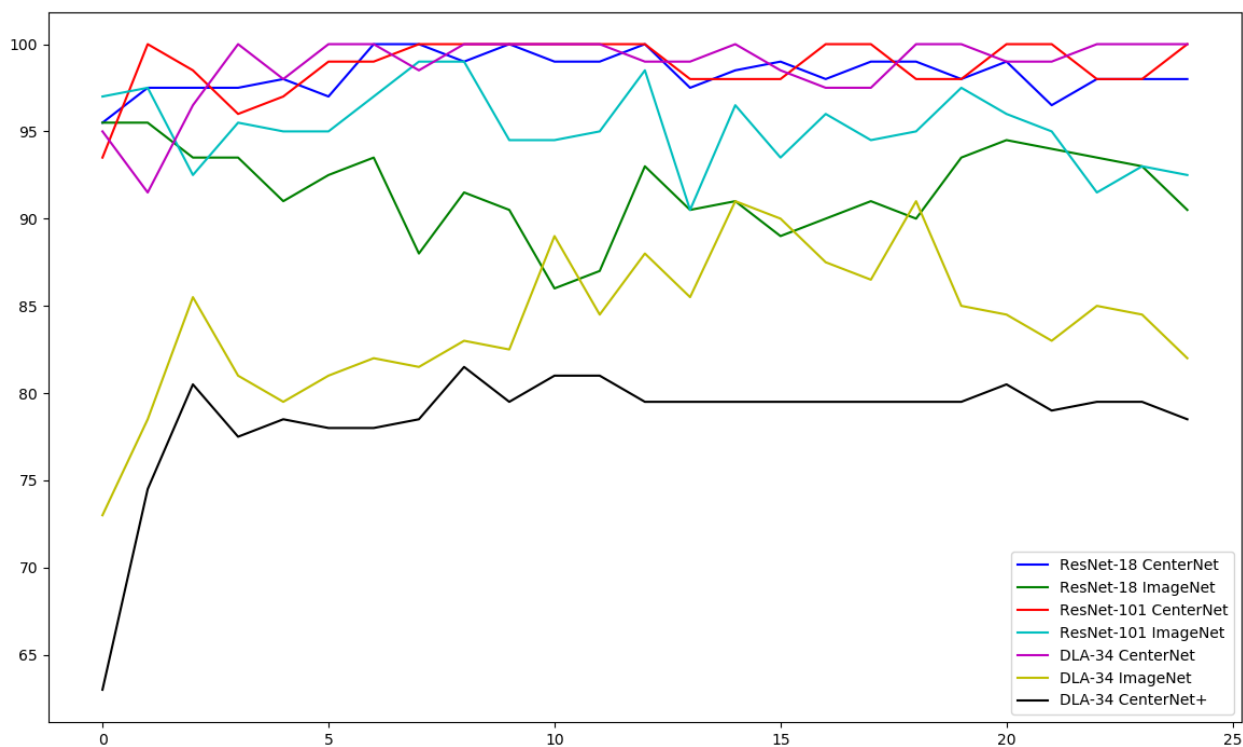
6 lentelė. Modelių su skirtingais svoriais tikslumas mokymo epochose.

Tinklo modelis / Epocha	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
ResNet-18 CenterNet	0.955	0.975	0.975	0.975	0.980	0.970	1.000	1.000	0.990	1.000	0.990	0.990	1.000	0.975	0.985	0.990	0.980	0.990	0.990	0.980	0.990	0.965	0.980	0.980	0.980
ResNet-18 ImageNet	0.955	0.955	0.935	0.935	0.910	0.925	0.935	0.880	0.915	0.905	0.860	0.870	0.930	0.905	0.910	0.890	0.900	0.910	0.900	0.935	0.945	0.940	0.935	0.930	0.905
ResNet-101 CenterNet	0.935	1.000	0.985	0.960	0.970	0.990	0.990	1.000	1.000	1.000	1.000	1.000	1.000	0.980	0.980	0.980	1.000	1.000	0.980	0.980	1.000	1.000	0.980	0.980	1.000
ResNet-101 ImageNet	0.970	0.975	0.925	0.955	0.950	0.950	0.970	0.990	0.990	0.945	0.945	0.950	0.985	0.905	0.965	0.935	0.960	0.945	0.950	0.975	0.960	0.950	0.915	0.930	0.925
DLA-34 CenterNet	0.950	0.910	0.965	1.000	0.980	1.000	1.000	0.985	1.000	1.000	1.000	1.000	0.990	0.990	1.000	0.985	0.975	0.970	1.000	1.000	0.990	0.990	1.000	1.000	1.000
DLA-34 ImagenetNet	0.730	0.785	0.855	0.810	0.795	0.810	0.820	0.815	0.830	0.825	0.890	0.845	0.880	0.855	0.910	0.900	0.875	0.865	0.910	0.850	0.845	0.830	0.850	0.845	0.820
DLA-34 CenterNet+	0.630	0.745	0.805	0.775	0.785	0.780	0.780	0.785	0.815	0.795	0.810	0.810	0.795	0.795	0.795	0.795	0.795	0.795	0.795	0.795	0.805	0.790	0.795	0.795	0.785

2.3.2.1. Tikslumas

Lentelėje nr. 6 matyti, jog net trims tinklo modeliams pavyko pasiekti 1.00 (100%) tikslumą (10). Visi trys modeliai: ResNet-18 CenterNet, ResNet-101 CenterNet ir DLA-34 CenterNet naudojo tuos pačius, CenterNet tinklo svorius. Tai buvo galima nuspėti, nes naudojami tinklo modeliai paimti iš CenterNet sistemos. Nei vienam tinklo modeliui naudojančiam ImageNet svorius nepavyko pasiekti geriausio rezultato. Iš tinklų naudojančių ImageNet svorius geriausius rezultatus pademonstravo ResNet-101 ImageNet modelis, pasiekdamas 0.99 (99%) tikslumą aštuntoje epochoje. Kiti toki modeliai taip pat pasiekė gerą tikslumą > 0.90 . Prasčiausius rezultatus

parodė DLA-34 CenterNet+ tinklo modelis, nors buvo manyta, kad taikant išankstinį mokymą, tinklo modelis bus vienas geriausių (skyrus 2.3.1). Matyt išankstinio mokymo metu tinklas persimokė ir vėliau „užšaldžius“ bazinius sluoksnius nepavyksta pasiekti gero tikslumo. Paveikslėlyje nr. 24 vaizduojama tinklo modelių tikslumo (lentelės nr. 6) grafikas. Grafike matyti, kaip skiriasi tinklų tikslumai naudojant CenterNet, ImageNet ir CenterNet+ svorius.



24 pav. Modelių su skirtingais tinklo svoriais tikslumo vizualizacija.

2.3.2.2. Vidutinis tikslumas

Jeigu tikslumas (10) matuoja poros vaizdų $d(a, p)$ ir $d(a, n)$ rezultatą, tai vidutinis tikslumas (13) rodo vaizdų poros $d(a, p)$ santykį su visais testavimo duomenimis. Bendras visų testuojamų modelių (lentelė nr. 5) vidutinis tikslumas matomas lentelėje nr. 7.

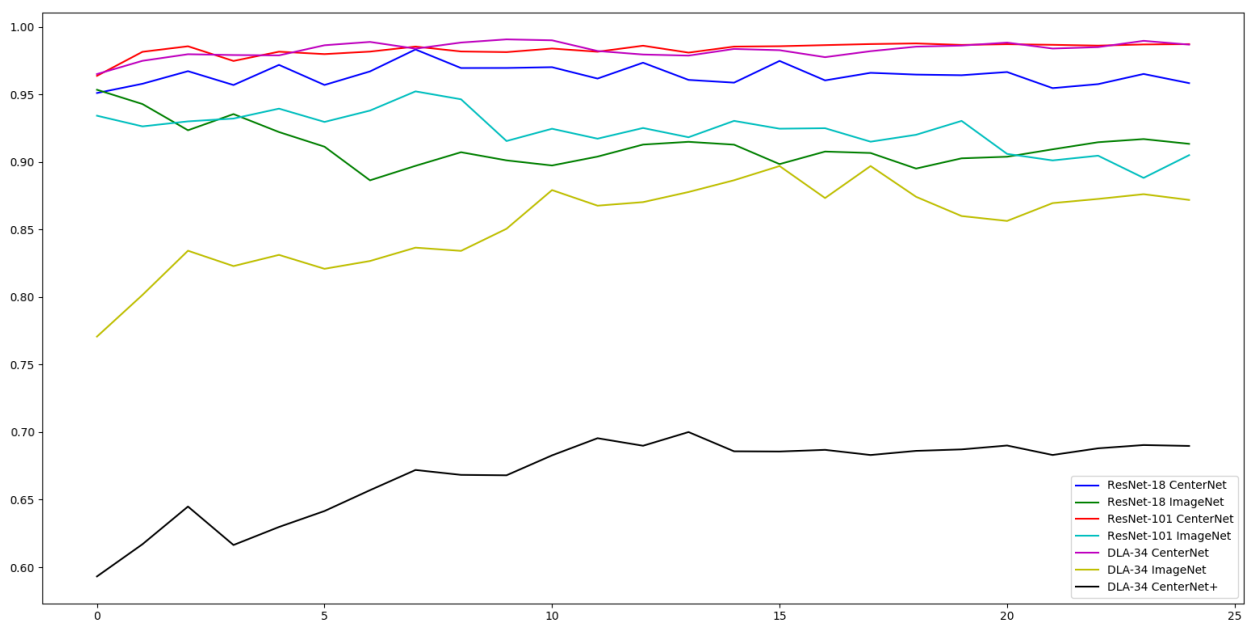
Geriausią vidutinį tikslumą 0.987 pavyko pasiekti tinklo modeliui ResNet-101 CenterNet. Iš grafiko (pav. 25) vaizduojančio lentelės nr. 7 duomenis matyti, kad DLA-34 CenterNet vidutinis tikslumas yra labai artimas geriausiems rezultatams parodžiusiam ResNet-101 CenterNet tinklo modeliui. Šiek tiek prastesnis vidutinis tikslumas yra ResNet-18 CenterNet tinklo modelio.

Kaip ir tikslumo rezultatuose (paveikslėlis nr. 24, lentelė nr. 6) geriausiems rezultatams parodė tinklo modeliai naudojantys CenterNet svorius, prastesni rezultatai yra tinklo modelių naudojusių ImageNet svorius. DLA-34 CenterNet+ pademonstravo prasčiausius vidutinio tikslumo rezultatus, kaip ir tikslumo metrikoje.

Lentelėje nr. 7 matyti, kad nei vienam tinklo modeliui nepavyko pasiekti 100% vidutinio tikslumo rezultato. Taip yra dėl duomenų rinkinio broko, kuris buvo pastebėtas jau atlikus eksperimentus. Kai kurie neigiami poveikslėliai n yra identiški kuriam nors kito tripleto baziniam poveikslėliui a . Dėl to unikalių testavimo vaizdų skaičius yra apie keturis kartus mažesnis.

7 lentelė. Modelių su skirtingais svoriais vidutinis tikslumas mokymo epochose.

Tinklo modelis / Epocha	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
ResNet-18 CenterNet	0.951	0.957	0.967	0.957	0.971	0.957	0.967	0.983	0.969	0.969	0.970	0.961	0.973	0.960	0.958	0.974	0.960	0.966	0.964	0.964	0.966	0.954	0.957	0.965	0.958
ResNet-18 ImageNet	0.953	0.943	0.924	0.935	0.922	0.911	0.886	0.897	0.907	0.901	0.897	0.904	0.913	0.915	0.913	0.898	0.908	0.907	0.895	0.903	0.904	0.909	0.915	0.917	0.913
ResNet-101 CenterNet	0.963	0.981	0.985	0.974	0.981	0.979	0.981	0.985	0.981	0.981	0.984	0.981	0.986	0.980	0.985	0.985	0.986	0.987	0.987	0.986	0.987	0.986	0.986	0.987	0.987
ResNet-101 ImageNet	0.934	0.926	0.930	0.932	0.940	0.930	0.938	0.952	0.946	0.916	0.925	0.917	0.925	0.918	0.930	0.925	0.925	0.915	0.920	0.930	0.906	0.901	0.905	0.888	0.905
DLA-34 CenterNet	0.965	0.974	0.979	0.979	0.978	0.986	0.988	0.983	0.988	0.991	0.990	0.982	0.979	0.978	0.983	0.982	0.977	0.982	0.985	0.986	0.988	0.983	0.985	0.989	0.986
DLA-34 ImagenetNet	0.593	0.617	0.644	0.616	0.629	0.641	0.657	0.672	0.668	0.668	0.682	0.695	0.689	0.700	0.685	0.685	0.686	0.683	0.686	0.687	0.690	0.683	0.688	0.690	0.689
DLA-34 CenterNet+	0.770	0.801	0.834	0.822	0.831	0.820	0.826	0.836	0.834	0.850	0.879	0.867	0.870	0.877	0.886	0.897	0.873	0.897	0.874	0.860	0.856	0.869	0.872	0.876	0.871



25 pav. Modelių su skirtingais tinklo svoriais vidutinio tikslumo vizualizacija.

2.3.2.3. Pearsono koreliacija

Pearsono koreliacijos tarp vaizdų porų (15) pateiktos lentelėje nr. 8. Lentelės vizualizacija pavaizduota paveikslėlyje nr. 24.

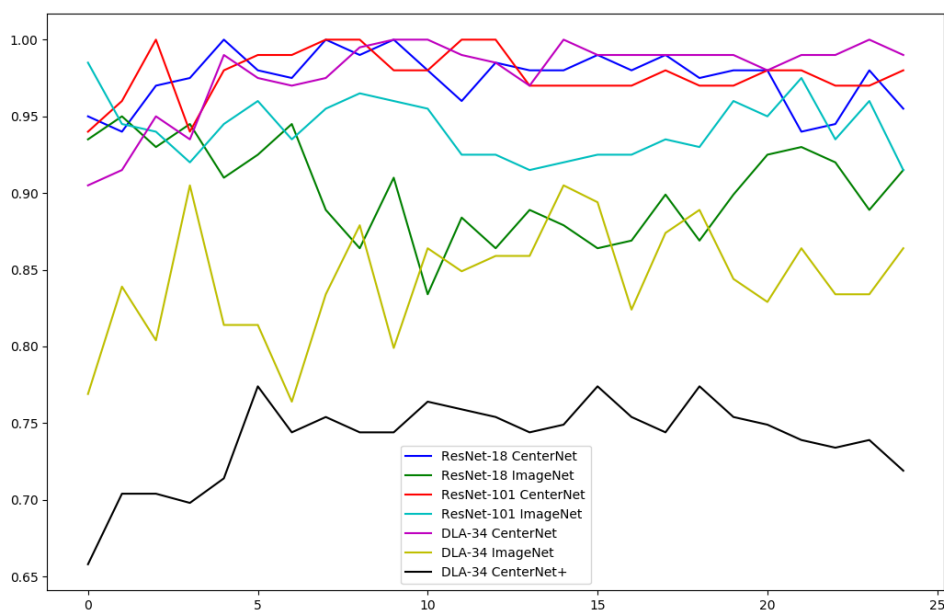
Pearsono koreliacija, kaip ir prieš tai naudotos kitos dvi metrikos davė panašius rezultatus. Tinklo modeliai naudojantys CenterNet svorius pademonstravo geriausias rezultatus, pasiekdami 100 % tikslumą. ImageNet svorių rezultatai yra prastesni, geriausias rezultatus su šiais svoriais pasiekė ResNet-101 ImageNet tinklas - 0.985. DLA-34 CenterNet+ modelio rezultatas, kaip ir kitose metrikose yra prasčiausias.

Toliau galima apžvelgti konkrečių modelių Pearsono koreliacijos rezultatus. Paimkime tris geriausias rezultatus parodžiusius skirtingos architektūros tinklo modelius ResNet-18 CenterNet (pav. nr. 27), ResNet-101 CenterNet (pav. nr. 28) ir DLA-34 CenterNet (pav. nr. 29).

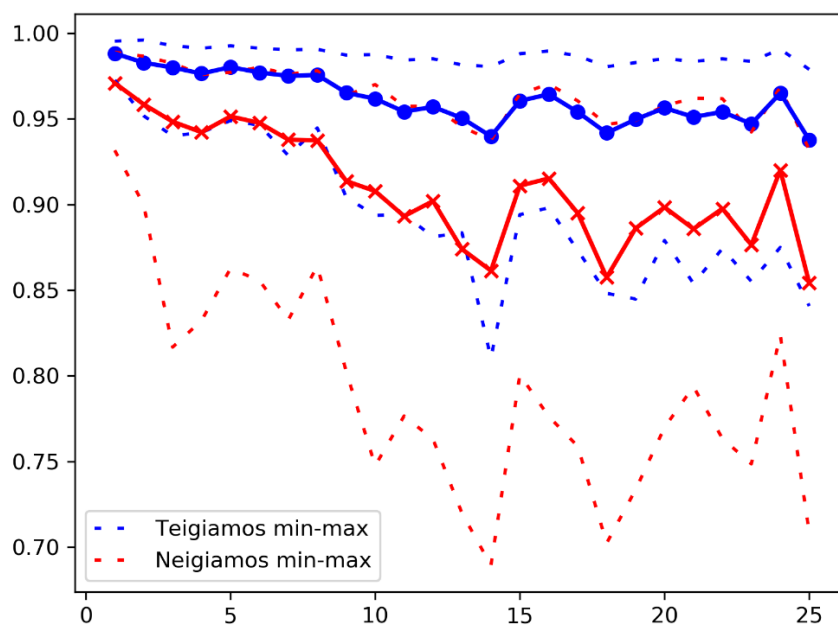
Visuose vaizduose mėlyna linija vaizduojama $preason(a,p)$ epochų vidurkius, raudona $preason(a,n)$ epochų vidurkius. Atitinkamai punktyrinės linijos yra minimumo ir maksimumo reikšmės.

8 lentelė. Modelių su skirtingais svoriais Pearsono koreliacijos mokymo epochose.

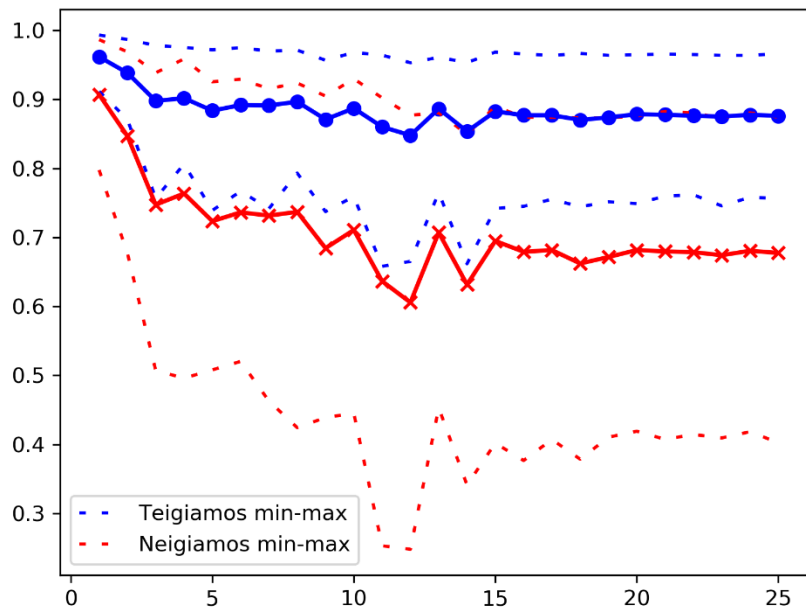
Tinklo modelis / Epocha	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	
ResNet-18 CenterNet	0.950	0.940	0.970	0.975	1.000	0.980	0.975	1.000	0.990	1.000	0.980	0.960	0.985	0.980	0.980	0.990	0.980	0.990	0.975	0.980	0.980	0.940	0.945	0.980	0.955	
ResNet-18 ImageNet	0.935	0.950	0.930	0.945	0.910	0.925	0.945	0.889	0.864	0.910	0.834	0.884	0.864	0.889	0.879	0.864	0.869	0.899	0.869	0.899	0.925	0.930	0.920	0.889	0.915	
ResNet-101 CenterNet	0.940	0.960	1.000	0.940	0.980	0.990	0.990	1.000	1.000	0.980	0.980	1.000	1.000	0.970	0.970	0.970	0.970	0.980	0.970	0.970	0.970	0.980	0.970	0.970	0.980	
ResNet-101 ImageNet	0.985	0.945	0.940	0.920	0.945	0.960	0.935	0.955	0.965	0.960	0.955	0.925	0.925	0.915	0.920	0.925	0.925	0.935	0.930	0.960	0.950	0.975	0.935	0.960	0.915	
DLA-34 CenterNet	0.905	0.915	0.950	0.935	0.990	0.975	0.970	0.975	0.995	1.000	1.000	0.990	0.985	0.970	1.000	0.990	0.990	0.990	0.990	0.990	0.990	0.980	0.990	0.990	1.000	0.990
DLA-34 ImagenetNet	0.769	0.839	0.804	0.905	0.814	0.814	0.764	0.834	0.879	0.799	0.864	0.849	0.859	0.859	0.905	0.894	0.824	0.874	0.889	0.844	0.829	0.864	0.834	0.834	0.864	
DLA-34 CenterNet+	0.658	0.704	0.704	0.698	0.714	0.774	0.744	0.754	0.744	0.744	0.764	0.759	0.754	0.744	0.749	0.774	0.754	0.744	0.774	0.754	0.749	0.739	0.734	0.739	0.719	



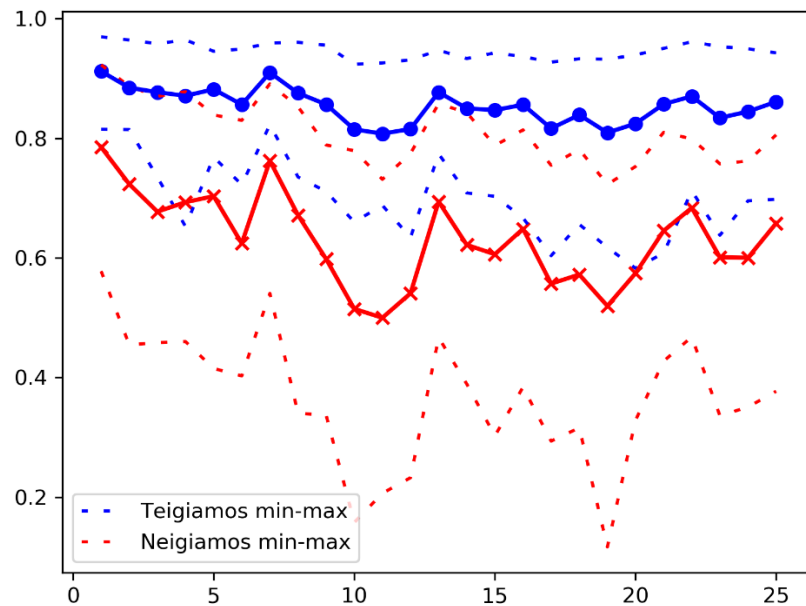
26. pav. Modelių su skirtingais tinklo svoriais Pearsono koreliacijos vizualizacija.



27 pav. ResNet-18 CenterNet Pearsono koreliacijos epochose.



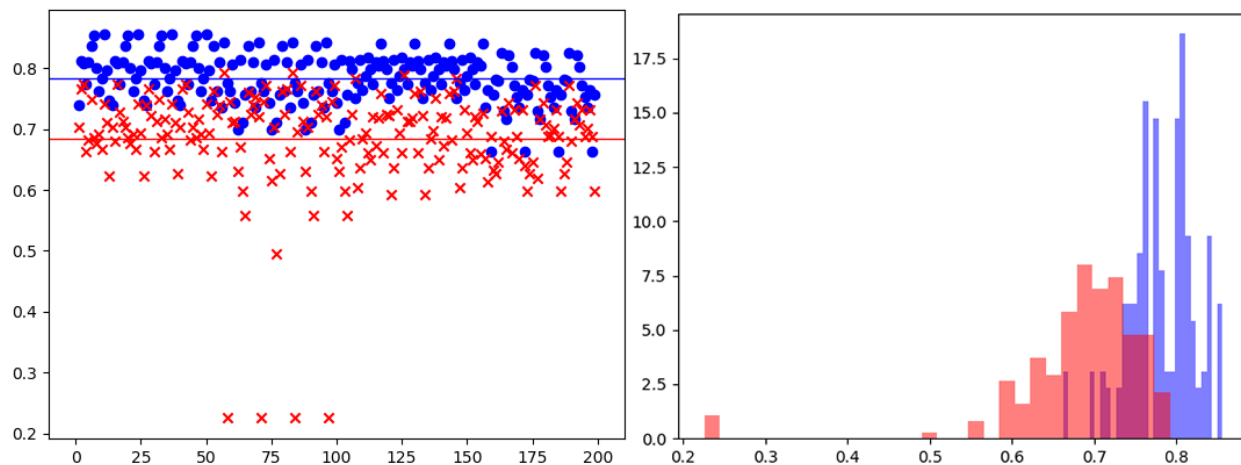
28 pav. ResNet-101 CenterNet Pearsono koreliacijos epochose.



29 pav. DLA-34 CenterNet Pearsono koreliacijos epochose.

Iš pateiktų grafikų matyti, kad teigiamos Pearsono koreliacijos reikšmės svyruoja diapazone tarp 0.8 ir 1.0. ResNet-101 CenterNet modelyje (pav. nr. 28) matoma, kaip po 15 epochos koreliacijos reikšmės nusistovi ir toliau svyruoja labai nedaug. Galime atlikti pjūvį minėtuose duomenyse, kad pamatyti, kaip atrodo individualios reikšmės. Pjūviui pasirinkta 20 epocha (pav. nr. 30). Kairiame vaizde matoma individualūs kiekvienos poros Pearsono koreliacijos rezultatai. Mėlyna spalva žymime $pearson(a,p)$ porą, raudona $pearson(a,n)$ porą. Dešinėje pusėje yra tų duomenų histograma. Joje galima išžvelgti korelacių persidengimą. Tačiau nėra nei vienos $pearson(a,n)$ poros, kurios koreliacijos reikšmė didesnė nei 0.807. Parsirinkus

koreliacijos reikšmę - 0.826 modelis 20 epochoje pasiekia 82.5 % teigiamų reikšmių tikslumą (16) ir 3.5% iš neigiamų reikšmių tenkina sąlygą (17).



30 pav. ResNet-101 CenterNet 20 epochos pjūvis.

2.4. Eksperimentai su tripleto tikslo funkcijomis

2.6 skyriuje apžvelgiamos trys tripleto tikslo funkcijos: standartinė (3), su pakeitimais (*angl. swap*) ir be nuostolių (*angl. lossless*) (4). Su šiomis tikslo funkcijomis ir bus atliekami eksperimentai. Iš tinklo svorių eksperimento (skyrius 3.3) nustatyta, kad CenterNet tinklo svoriai yra tinkamiausi ortofotografinių vaizdų panašumo mokymui, todėl šiame eksperimente tik jie ir bus naudojami. Tikslas pamatyti, kaip tripleto tikslo funkcija gali įtakoti tinklo tikslumą ir mokymąsi ir nustatyti kuri iš tikslo funkcijų duoda geriausius rezultatus. Standartinė ir funkcija su pakeitimais yra įgyvendinta kaip Pytorch bibliotekos dalis (Class TripletMarginLoss¹⁸), jos bus ir naudojamos. Funkcija be nuostolių įgyvendinama pagal formulę (4). Eksperimento tinklų modelių matrica matoma lentelėje nr. 10.

10 lentelė. Tripleto tikslo funkcijų eksperimentų matrica.

Tinklas \ Funkcija	Standartinė (S)	Su pakeitimais (SW)	Be nuostolių (L)
ResNet-18	ResNet-18 S	ResNet-18 SW	ResNet-18 L
ResNet-101	ResNet-101 S	ResNet-101 SW	ResNet-101 L
DLA-34	DLA-34 S	DLA-34 SW	DLA-34 L

¹⁸ https://pytorch.org/cppdocs/api/structtorch_1_1nn_1_1_triplet_margin_loss_impl.html

2.4.1. Rezultatai

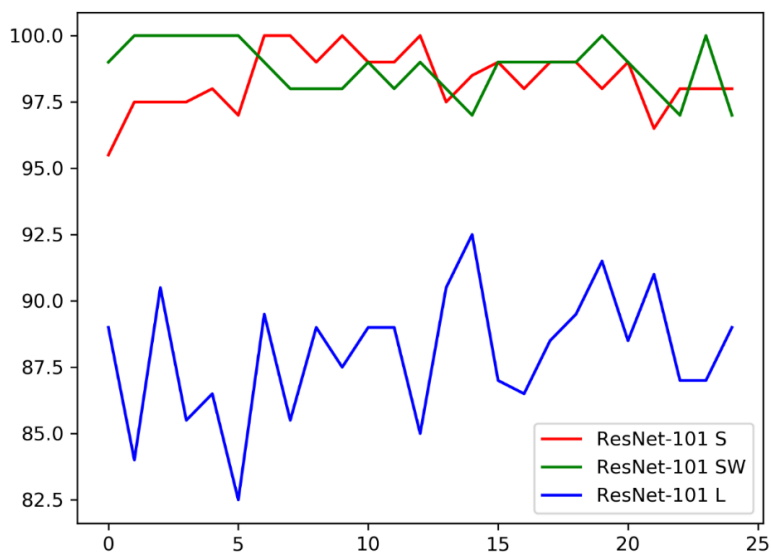
Žalsva spalva žymimi geriausi lentelės rezultatai, pažymimi visi modeliai ir epochos kuriems pavyko pasiekti geriausią rezultatą. Rausva spalva pažymima geriausias konkretaus modelio rezultatas, jeigu jis nėra geriausiai tos lentelės rezultatas.

2.4.1.1. Tikslumas

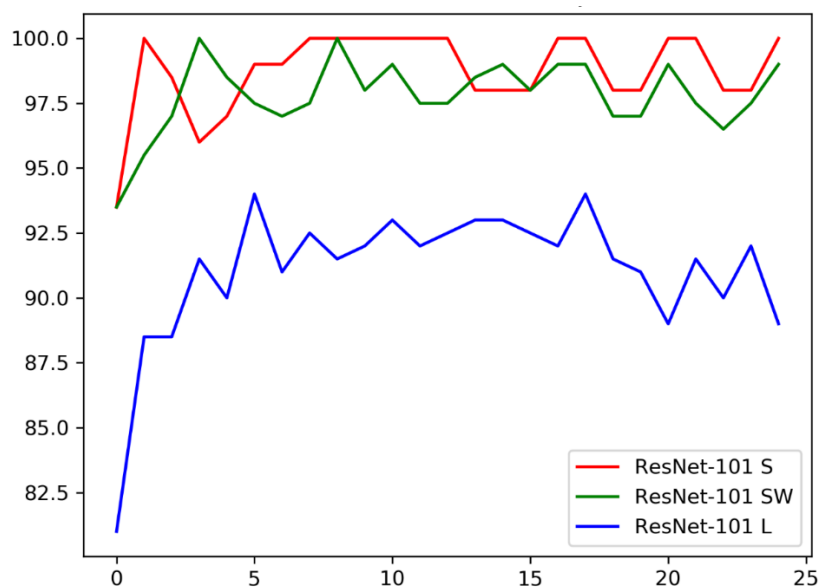
Lentelėje nr. 11 vaizduojami visų eksperimento modelių (lentelė nr. 10) tikslumo (10) vidurkiai 25 epochose. Visiems tinklo modeliams, išskyrus naudojusiems tirpleto tikslo funkcija be nuostolių, pavyko pasiekti 100% tikslumą. Paveikslėlyje nr. 31 vaizduojama ResNet-18 tinklo visų tiriamų tikslo funkcijų grafikas. Tinklo modelis naudojantis tirpleto tinklo funkcija su pakeitimais (ResNet-18 SW) sugebėjo greičiausiai pasiekti tiksliausią rezultatą. Jau po antros epochos. Nors ir ResNet-101 tinklas dalinasi ta pačia architektūra, kaip ir ResNet-18, skiriasi tik jų gylis, tačiau tinklo funkcija su pakeitimais jame nedavė tokių rezultatų kaip mažesnėje (ResNet-18 SW) tinklo versijoje. ResNet-101 tinklo modeliai pavaizduoti paveikslėlyje nr. 32. Tirpleto tinklo funkcija su be nuostolių (4) visais atvejais parodė prasčiausius rezultatus lyginant su vienodos architektūros modeliais.

11 lentelė. Tirpleto tikslo funkcijų eksperimentų matrica.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
ResNet-18 S	0.955	0.975	0.975	0.975	0.980	0.970	1.000	1.000	0.990	1.000	0.990	0.990	1.000	0.975	0.985	0.990	0.980	0.990	0.990	0.980	0.990	0.965	0.980	0.980	0.980
ResNet-18 SW	0.990	1.000	1.000	1.000	1.000	1.000	0.990	0.980	0.980	0.980	0.990	0.980	0.990	0.980	0.970	0.990	0.990	0.990	0.990	1.000	0.990	0.980	0.970	1.000	0.970
ResNet-18 L	0.890	0.840	0.905	0.855	0.865	0.825	0.895	0.855	0.890	0.875	0.890	0.890	0.850	0.905	0.925	0.870	0.865	0.885	0.895	0.915	0.885	0.910	0.870	0.870	0.890
ResNet-101 S	0.935	1.000	0.985	0.960	0.970	0.990	0.990	1.000	1.000	1.000	1.000	1.000	1.000	0.980	0.980	0.980	1.000	1.000	0.980	0.980	1.000	1.000	0.980	0.980	1.000
ResNet-101 SW	0.935	0.955	0.970	1.000	0.985	0.975	0.970	0.975	1.000	0.980	0.990	0.975	0.975	0.985	0.990	0.980	0.990	0.990	0.990	0.970	0.970	0.990	0.975	0.965	0.975
ResNet-101 L	0.810	0.885	0.885	0.915	0.900	0.940	0.910	0.925	0.915	0.920	0.930	0.920	0.925	0.930	0.930	0.925	0.920	0.940	0.915	0.910	0.890	0.915	0.900	0.920	0.890
DLA-34 S	0.950	0.910	0.965	1.000	0.980	1.000	1.000	0.985	1.000	1.000	1.000	1.000	0.990	0.990	1.000	0.985	0.975	0.970	1.000	1.000	0.990	0.990	1.000	1.000	1.000
DLA-34 SW	0.925	0.955	0.975	0.960	0.980	0.970	0.985	1.000	0.985	0.985	1.000	1.000	0.965	0.985	0.980	0.990	0.980	1.000	0.990	0.995	0.995	0.995	0.995	0.995	1.000
DLA-34 L	0.890	0.840	0.905	0.855	0.865	0.825	0.895	0.855	0.890	0.875	0.890	0.890	0.850	0.905	0.925	0.870	0.865	0.885	0.895	0.915	0.885	0.910	0.870	0.870	0.890



31 pav. Resnet-18 tikslo funkcijų tikslumo palyginimas.



32 pav. Resnet-101 tikslo funkcijų tikslumo palyginimas.

2.4.1.2. Vidutinis tikslumas

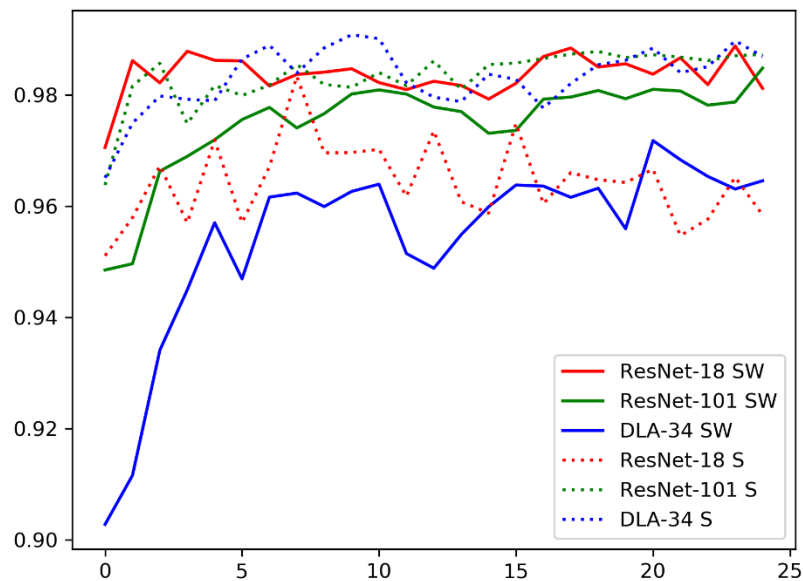
Vidutinio tikslumo (13) eksperimentų modelių rezultatai pavaizduoti lentelėje nr. 12. Nors didžiausią tikslumą pasiekė DLA-34 S tinklo modelis, tačiau kitų geriausių modelių vidutinis tikslumas mažesnis vos keliomis tūkstantosiomis dalimis. Jeigu tikslumo metrikoje (10) vaizduojami paveikslėliai (pav. nr. 31 ir pav. nr. 32) pagal tinklų architektūrą, vidutinio tikslumo metrikoje vaizduokime rezultatus iš tikslo funkcijų pusės.

12 lentelė. Tikslo funkcijų vidutinis tikslumas

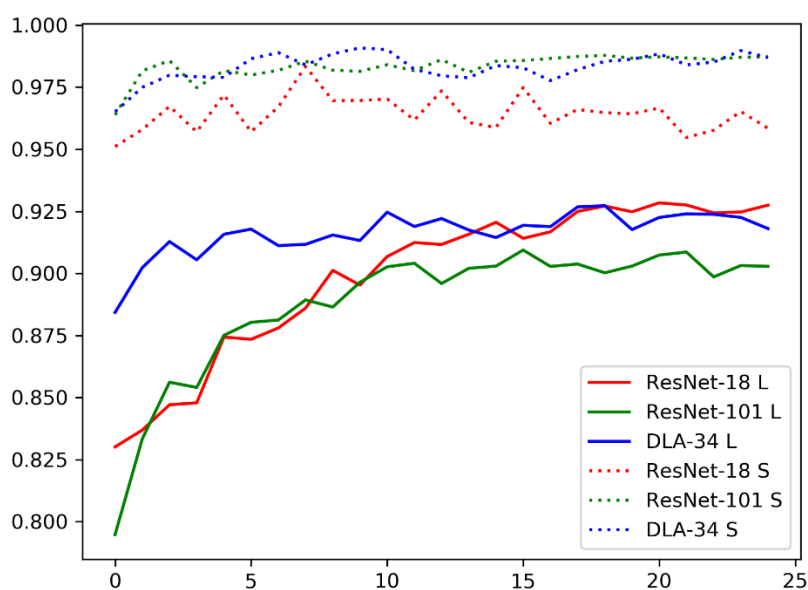
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
ResNet-18 S	0.951	0.957	0.967	0.957	0.971	0.957	0.967	0.983	0.969	0.969	0.970	0.961	0.973	0.960	0.958	0.974	0.960	0.966	0.964	0.964	0.966	0.954	0.957	0.965	0.958
ResNet-18 SW	0.971	0.986	0.982	0.988	0.986	0.986	0.982	0.984	0.984	0.985	0.982	0.981	0.982	0.982	0.979	0.982	0.987	0.988	0.985	0.986	0.984	0.987	0.982	0.989	0.981
ResNet-18 L	0.830	0.837	0.847	0.848	0.874	0.873	0.878	0.886	0.901	0.895	0.907	0.912	0.912	0.916	0.921	0.914	0.917	0.925	0.927	0.925	0.928	0.928	0.924	0.925	0.927
ResNet-101 S	0.963	0.981	0.985	0.974	0.981	0.979	0.981	0.985	0.981	0.981	0.984	0.981	0.986	0.980	0.985	0.985	0.986	0.987	0.987	0.986	0.987	0.986	0.986	0.987	0.987
ResNet-101 SW	0.949	0.950	0.966	0.969	0.972	0.976	0.978	0.974	0.977	0.980	0.981	0.980	0.978	0.977	0.973	0.974	0.979	0.980	0.981	0.979	0.981	0.981	0.978	0.979	0.985
ResNet-101 L	0.795	0.833	0.856	0.854	0.875	0.880	0.881	0.889	0.886	0.896	0.903	0.904	0.896	0.902	0.903	0.909	0.903	0.904	0.900	0.903	0.907	0.909	0.899	0.903	0.903
DLA-34 S	0.965	0.974	0.979	0.979	0.978	0.986	0.988	0.983	0.988	0.991	0.990	0.982	0.979	0.978	0.983	0.982	0.977	0.982	0.985	0.986	0.988	0.983	0.985	0.989	0.986
DLA-34 SW	0.903	0.912	0.934	0.945	0.957	0.947	0.962	0.962	0.960	0.963	0.964	0.951	0.949	0.955	0.960	0.964	0.964	0.962	0.963	0.956	0.972	0.968	0.965	0.963	0.965
DLA-34 L	0.884	0.902	0.913	0.905	0.916	0.918	0.911	0.912	0.915	0.913	0.925	0.919	0.922	0.917	0.914	0.919	0.919	0.927	0.927	0.918	0.923	0.924	0.924	0.923	0.918

Paveikslėlyje nr. 33 matyti tripleto tikslo funkcijos su pakeitimais (vientisa linija) ir bazinės tikslo funkcijos (brūkšninė linija) vidutinio tikslumo rezultatai. ResNet-101 S ir DLA-34 S tinklo modeliai rodo geresnius rezultatus, nei jų atitikmenys naudojantys tikslo funkciją su pakeitimais. ResNet-18 tinklo modeliai (S ir SW) veikia priešingai. ResNet-18 SW yra gerokai tikslesnis nei ResNet-18 S. Galima daryti išvada, kad mažesniems tinklams naudoti tikslo funkcija su pakeitimais yra efektyviau.

Paveikslėlyje nr. 34 matyti tripleto tikslo funkcijos be nuostolių (vientisa linija) ir bazinės tikslo funkcijos (brūkšninė linija) vidutinio tikslumo rezultatai. Tikslo funkcijos be nuostolių vidutinio tikslumo rezultatai yra gerokai prastesni, nei bazinės tikslo funkcijos.



33 pav. SW ir S tikslo funkcijų vidutinis tikslumas.



34 pav. L ir S tikslo funkcijų vidutinis tikslumas.

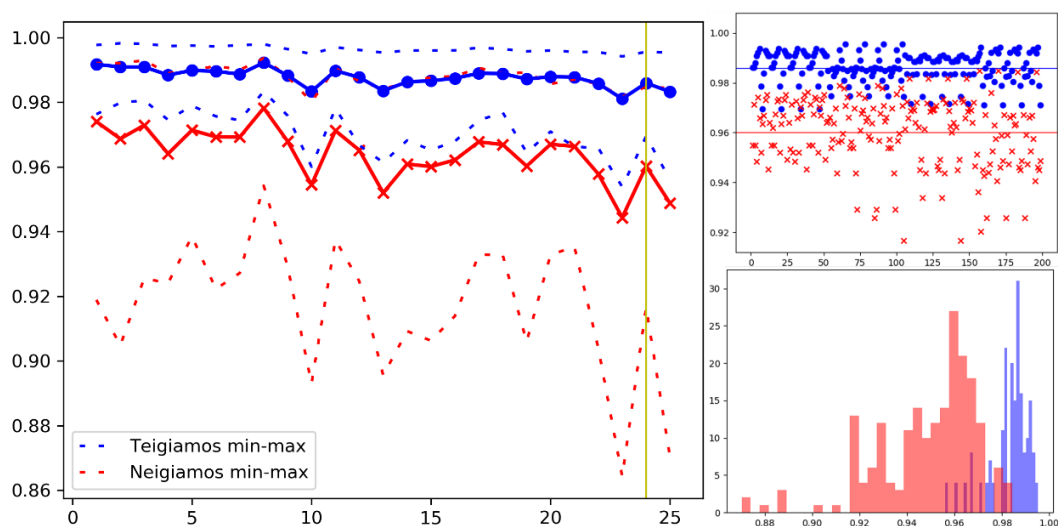
2.4.1.3. Pearsono koreliacija

Pearsono koreliacijos tarp vaizdų porų (15) pateiktos lentelėje nr. 13. Lentelė duoda panašų vaizdą kaip ir tikslumo metrikos (10) lentelė nr. 11. ResNet-18 SW tinko modelis pasiekia 100% Pearsono tikslumo rezultatą anksčiausiai. Norint susidaryti vaizdą apie rezultatus, vizualizuokime kelių pasirinktų modelių rezultatus su duomenų pjūviais.

13 lentelė. Pearsono koreliacijos pagal skirtingas tripleto tikslo funkcijas.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
ResNet-18 S	0.950	0.940	0.970	0.975	1.000	0.980	0.975	1.000	0.990	1.000	0.980	0.960	0.985	0.980	0.980	0.990	0.980	0.990	0.975	0.980	0.980	0.940	0.945	0.980	0.955
ResNet-18 SW	0.990	1.000	1.000	1.000	1.000	0.990	0.990	0.980	0.990	0.980	0.990	0.980	0.990	0.980	0.980	0.975	0.990	0.990	0.990	1.000	0.990	0.970	0.970	0.990	0.970
ResNet-18 L	0.809	0.884	0.905	0.915	0.920	0.935	0.910	0.925	0.915	0.920	0.930	0.920	0.945	0.930	0.930	0.925	0.920	0.940	0.915	0.910	0.889	0.915	0.905	0.920	0.889
ResNet-101 S	0.940	0.960	1.000	0.940	0.980	0.990	0.990	1.000	1.000	0.980	0.980	1.000	1.000	0.970	0.970	0.970	0.970	0.980	0.970	0.970	0.980	0.980	0.970	0.970	0.980
ResNet-101 SW	0.930	0.955	0.970	0.980	0.965	0.970	0.930	0.970	0.985	0.960	0.935	0.955	0.960	0.975	0.970	0.970	0.970	0.940	0.950	0.970	0.940	0.965	0.960	0.950	0.950
ResNet-101 L	0.809	0.884	0.905	0.915	0.920	0.935	0.910	0.925	0.915	0.920	0.930	0.920	0.945	0.930	0.930	0.925	0.920	0.940	0.915	0.910	0.889	0.915	0.905	0.920	0.889
DLA-34 S	0.905	0.915	0.950	0.935	0.990	0.975	0.970	0.975	0.995	1.000	1.000	0.990	0.985	0.970	1.000	0.990	0.990	0.990	0.990	0.990	0.980	0.990	0.990	1.000	0.990
DLA-34 SW	0.970	0.975	0.950	0.965	0.980	0.945	0.955	0.945	0.965	0.945	0.965	0.955	0.960	0.960	0.970	0.990	0.975	0.980	0.965	0.955	0.975	0.955	0.955	0.955	0.990
DLA-34 L	0.905	0.940	0.935	0.955	0.960	0.960	0.955	0.955	0.945	0.965	0.975	0.975	0.965	0.960	0.960	0.980	0.975	0.975	0.975	0.955	0.950	0.960	0.940	0.935	0.955

Paveikslėlyje nr. 35 pavaizduota ResNet-18 SW modelio Pearsono koreliacijos tarp $pearson(a, p)$ mėlyna linija ir $pearson(a, n)$ raudona linija. Atitinkamai brūkšniuotos linijos yra koreliacijų minimumai ir maksimumai. $pearson(a, n)$ poros maksimumo linija yra artima. $pearson(a, p)$ poros vidurkiui. Panašų fenomeną galima stebėti ir ResNet-101 CenterNet modelyje (pav. nr. 28). Nuo pirmos epochos linijos nėra artimos minėtame modelyje, tačiau ties 10 epocha jos suvienodėja (bet nėra vienodos). Tokios situacijos nėra nei viename DLA-34 architektūros modelyje. Peržiūrėjus visus ResNet-18 ir ResNet-101 architektūrų modelius, nei viename $pearson(a, n)$ poros maksimumo linija nenusileidžia žemiau $pearson(a, p)$ poros vidurkiu. ResNet-18 S, ResNet-18 SW ir ResNet-101 S modeliuose šios linijos yra labai artimos viena kitai. Būtent šie modeliai ir prognozuoja geriausias rezultatus tarp ResNet architektūros modelių. Todėl galima teigti, kad suartėjus šitoms linijoms modelis pasiekia geriausias savo rezultatus. ResNet-18 S ir ResNet-18 SW apsimoko jau po pirmos epochos (matyt dėl savo mažesnio sluoksnių skaičiaus), ResNet-101 S po dešimtos epochos.

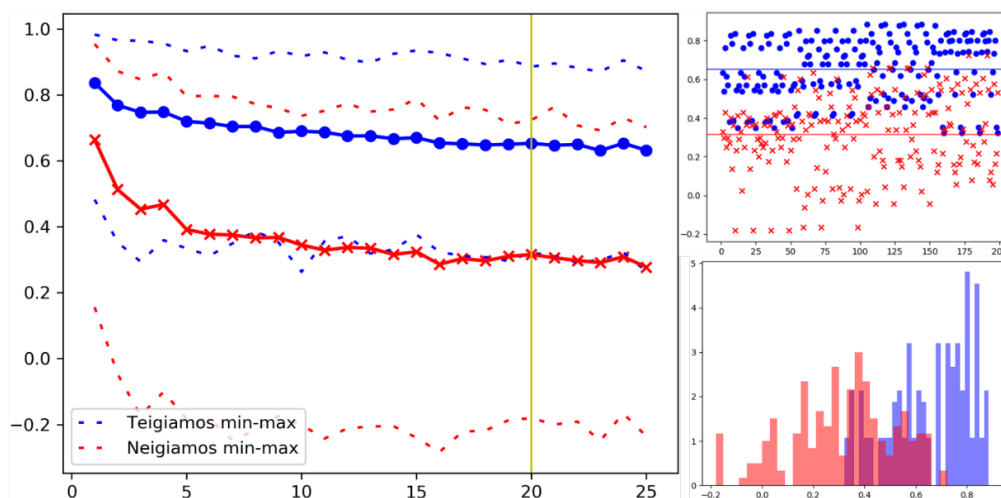


35 pav. ResNet-18 SW Pearsono koreliacijos ir pjūvis.

Paveikslėlyje nr. 35 pateikiamas ir 24-tos epochos ResNet-101 S pjūvis. Tokia epocha pasirinkta dėl vidutinio tikslumo rezultatų (lentelė nr. 12). Histogramoje matyti, kad persidengimas yra ties kitokiomis koreliacijos reikšmėmis nei ResNet-101 S modelio (pav. nr. 30). Galime rasti

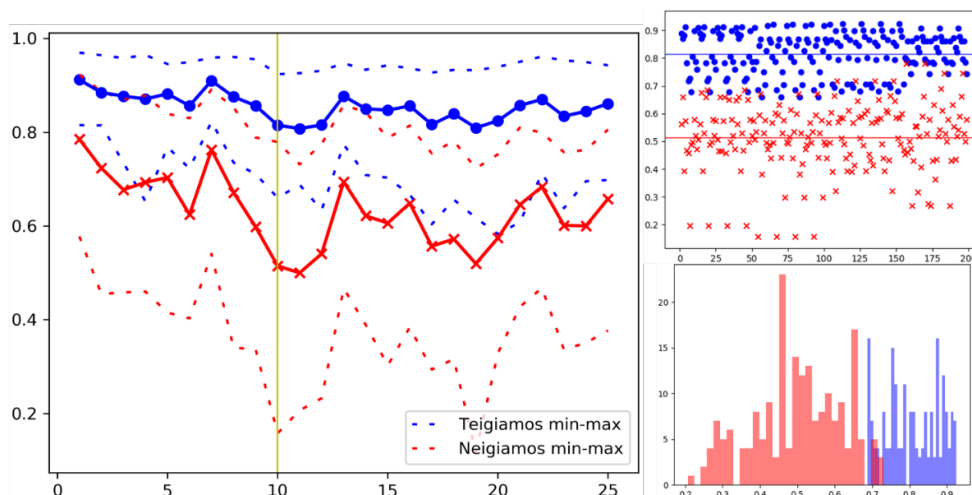
optimalią koreliacijos reikšmę, kada ResNet-18 SW geriausiai įvertina vaizdų vienodumą. Šiuo atveju esant Pearsono koreliacijos reikšmei 0.9768. 89,5% teigiamų vaizdų būtų laikomi panašiais (16) ir mažiau nei 1 % neigiamų vaizdų duotų teigiamą koreliacijos rezultatą (17).

Paveikslėlyje nr. 36 pavaizduota ResNet-101 L modelio Pearsono koreliacijos, žymėjimai naudojami toki patys kaip ir praeitame paveikslėlyje. Šituo atveju nematyti vidurkio ir maksimumo reikšmių persidengimo. Taip pat iš histogramos matosi daug daugiau persidengiančių reikšmių. Tripleto tikslo funkcija be liekanų nedemonstruoja pačių tiksliausių rezultatų. Naudojant Pearsono koreliacijos reikšmę 0.573 tik 66% teigiamų vaizdų yra identifikuojami kaip panašūs (16) ir 1,1 % neigiamų vaizdų duotų teigiamą rezultatą (17).



36 pav. ResNet-101 L Pearsono koreliacijos ir pjūvis.

Paveikslėlyje nr. 37 pavaizduota DLA-34 S modelio Pearsono koreliacijos ir pjūvis ties 10-ta epocha. DLA architektūros modeliuose kitaip nei ResNet architektūros modeliuose $pearson(a, n)$ poros maksimumo reikšmė nusileidžia žemiau nei $pearson(a, p)$ poros vidurkis. Todėl galima būtų teikti, kad šitas modelis geriausiai gali atskirti vaizdų panašumus, matuojant Pearsono koreliacija. Iš histogramos matyti, jog rezultatų persidengimas yra nedidelis. Esant Pearsono koreliacijai 0.629 net 99.5% teigiamų vaizdų yra identifikuojami kaip panašūs (16), ir mažiau nei 2% neigiamų vaizdų duos teigiamą panašumo rezultatą (17).



37 pav. DLA-34 S Pearsono koreliacijos ir pjūvis.

2.5. Palyginimas su VGG16

Vienintelis kitas tripleto tinklas vertinantis ortofotografinių vaizdų panašumus yra aprašytas Vilniaus universiteto mokslininkų publikuotame straipsnyje [VMR20]. Darbas apžvelgtas skyriuose 1.2 ir 1.4. Darbe siūlomas modifikuotas VGG16 tripleto tinklas vaizdų panašumui atpažinti. Remiantis straipsniu, atkurtas VGG16(0) modelis turintis 49 mokomus parametrus, parodęs geriausius rezultatus straipsnio tyrime.

Pagrindiniai skirtumai tarp autoriaus darbe naudojamų duomenų VGG16 tinklo mokymui ir CenterNet tinkle naudotų matomi lentelėje nr. 14. Tyrime rekomenduojama tinklą mokinti 16 epochų, tačiau dėl kelis kartus didesnio duomenų rinkinio, VGG16 buvo mokomas ir testuojamas 25 epochas, matuojant šiame darbe aprašytomis metrikomis. Dar vienas skirtumas tarp tinklų yra įvesties dydis. CenterNet tinklo įvesties vaizdo dydis yra 512x512 pikselių, VGG16 224x224 pikselių.

Darbe VGG16 tinklas buvo apmokytas tais pačiais svoriais kaip ir CenterNet, tik sumažinus vaizdų dydį iki 224x224 pikselių.

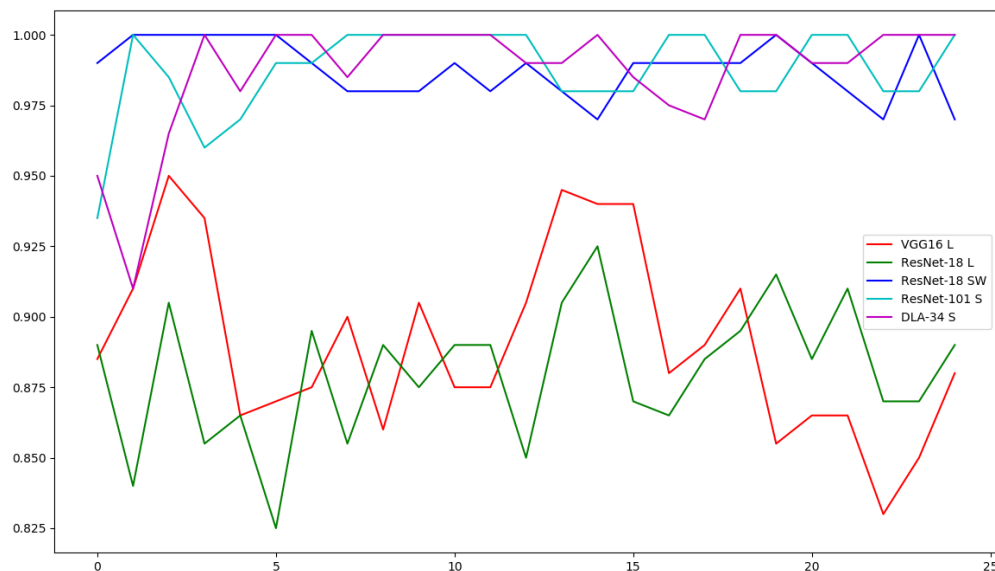
14 lentelė. VGG16 autoriaus ir CenterNet duomenų rinkinių skirtumai.

	VGG16	CenterNet
Mokymo vaizdų imtis	3,564	18200
Validacijos vaizdų imtis	315	1800
Testavimo vaizdų imtis	192	600

Su VGG16 tinklu (grafikuose žymimas kaip VGG16 L, L indikuoja kad tinklas naudoja tikslo funkciją be nuostolių (4)) palyginama CenterNet tinklo modeliai:

- ResNet-18 L. ResNet18 yra panašaus gylio tinklas kaip ir VGG16. L žymi, kad tinklas taip pat naudoja tripleto tikslo funkciją be nuostolių,
- ResNet-18 SW. Geriausių rezultatų parodęs ResNet-18 modelis. SW žymi, kad naudojama tripleto tikslo funkcija su pakeitimais,
- ResNet-101 S. Geriausių rezultatų parodęs ResNet-101 modelis. S žymi, kad naudojama standartinė tripleto tikslo funkcija (3),
- DLA-34 S. Geriausių rezultatų parodęs DLA-34 modelis. S žymi, kad naudojama standartinė tripleto tikslo funkcija.

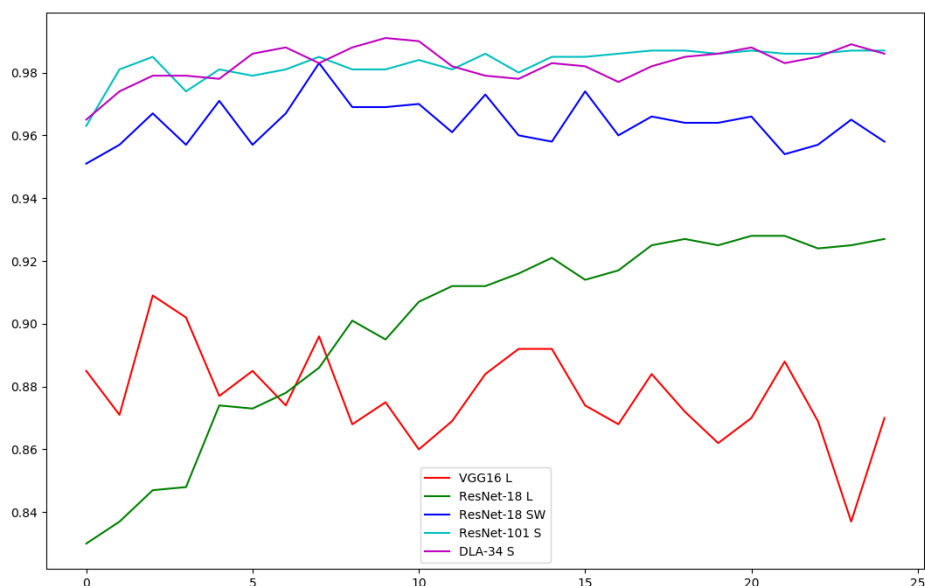
Paveikslėlyje nr. 38 matoma tinklų modelių tikslumo (10) metrikos rezultatai. Iš grafiko galima matyti, kad VGG16 L tikslumas šiek tiek didesnis, nei ResNet-18 L. Abu naudoja tą pačią tikslo funkciją. Visų kitų tinklų modelių tikslumas yra gerokai didesnis nei VGG16 L.



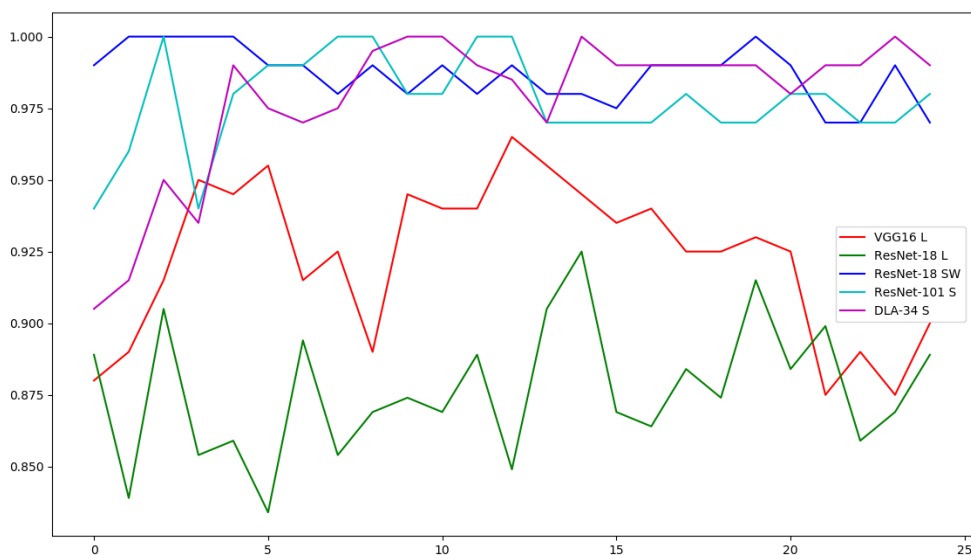
38 pav. Tikslumo palyginimas tarp CenterNet modelių VGG16.

Paveikslėlyje nr. 39 vaizduojama tinklai pagal vidutinio tikslumo (13) metriką. Šituo matavimu VGG16 demonstruoja prasčiausius rezultatus. Pagal ją galima spręsti, kad nors tinklas moka gerai nustatyti panašumą tarp dvejų porų $d(a, p)$ ir $d(a, n)$, bet bendrai visai sistemai tinklas prašiausiai įvertina panašumus tarp vaizdų.

Paveikslėlis nr. 40 vaizduoja Pearsono koreliaciją tarp vaizdų porų (15). VGG16 L demonstruoja geresnius rezultatus nei ResNet-18 L, bet nusileidžia visiems kitiems lyginamiems CenterNet tinklo modeliams.



39 pav. Vidutinio tikslumo palyginimas tarp CenterNet modelių ir VGG16.



39 pav. Vidutinio tikslumo palyginimas tarp CenterNet modelių ir VGG16.

3. Išvados

Pasiūlyta giliojo neuroninio tinklo CenterNet modifikacija ortofotografinių vaizdų atpažinimui, kuri leidžia efektyviai palyginti du žemėlapius. Siūloma modifikacija paremta tripleto tinklų architektūra. Ta pati modifikacija pritaikyta naudoti su trimis CenterNet baziniais tinklo modeliais: ResNet-18, ResNet-101 ir DLA-34.

Eksperimentams su tinklo svoriais buvo pasirinkti trys tinklų svoriai: CenterNet, ImageNet ir tarpinio mokymo tinklų svoriai. Eksperimentai parodė, kad CenterNet tinklo svoriai yra geriausi naudoti mokant tinklą vaizdų panašumui nustatyti. Visi tinklo modeliai naudojantys CenterNet svorius pasiekė 100% tikslumą. Didžiausias tikslumas naudojant ImageNet svorius 99%, buvo

pasiektas ResNet-101 tinklo modelio. Atitinkamai DLA-34 ir ResNet-18 pasiekė 95,5% ir 91,1% tikslumą. Lyginant tuos pačius tinklo svorius naudojant vidutinio tikslumo metriką geriausius rezultatus parodė DLA-34 su CenterNet tinklo svoriais - 99,1%. ResNet-101 su tais pačiais svoriais vidutinis tikslumas buvo 0,4% mažesnis, ResNet-18 0,8% mažesnis. Tarpinio mokymo svoriai, kurie buvo pasiūlyti darbe, eksperimento metu parodė prasčiausius rezultatus. 81,5% tikslumas ir 89,7% vidutinis tikslumas. Tai turbūt įvyko dėl tinklo persimokymo pirminio mokymo metu, kurią galėjo lemti per mažas mokymo duomenų kiekis ar įvairovė.

Tripleto tikslo funkcijų tyrimui buvo pasirinkta trys funkcijos: standartinė, su pakeitimais ir be nuostolių. Eksperimentai su tripleto tikslo funkcijomis parodė, kad esant mažesniam tinklui (ResNet-18) tinklo funkcija su pakeitimais greičiau apmoko tinklą, nei standartinė tripleto tikslo funkcija. ResNet-18 tinklo modelis 100% tikslumą naudodamas tinklo funkcija su pakeitimais jau pasiekė 2 epochoje, naudojant standartinę tripleto funkcija jam prireikė 7 epochų. Gilesnių tinklų atveju (ResNet-101 ir DLA-34) geresnius rezultatus rodo standartinė tripleto tikslo funkcija. Nors abejais tikslo funkcijų atvejais tinklo modeliams pavyko pasiekti 100% tikslumą, tačiau ResNet-101 su standartinė tripleto tikslo funkcija tokį tikslumą pasiekė jau 2 epochoje, o funkcijoje su pakeitimais 4 epochoje, atitinkamai DLA-34 4-toje ir 8-toje epochose. Tripleto tikslo funkcija be nuostolių parodė prasčiausius rezultatus eksperimente. Nei vienam iš modelių naudojant šią funkciją nepavyko pasiekti 100% tikslumo. Didžiausias tikslumas pasiektas ResNet-101 tinklo modelio - 94%. Galima daryti išvada, kad tai priklauso nuo paskutiniame sluoksnyje esančios sigmoidinės aktyvacijos funkcijos, kurios reikalauja funkcija. Funkcija be nuostolių buvo pasiūlyta naudoti prie siamo tinklų, kurie vertina ar vaizdas priklauso norimai klasei, kai tripleto tinklai lygina dvi vaizdų poras. Lyginant vaizdų panašumus yra svarbios ir neigiamos požymių vektorių reikšmės, nes ir neigiamas atstumas irgi yra atstumas, o jos išnyksta naudojant sigmoidinę aktyvacijos funkciją. Vertinant ortofotografinių vaizdų panašumus tripleto tikslo funkcija be nuostolių nėra siūloma naudoti.

Lyginant tikslo funkcijas su vidutinio tikslumo metrika, geriausią rezultatą pasiekė DLA-34 architektūros tinklas - 99,1 %, ResNet-101 ir ResNet-18 vidutiniai tikslumai - 98,7 % ir 98,9%. ResNet-18 pademonstravo geresnius rezultatus naudodamas tripleto funkciją su pakeitimais, nei gilesnis tokios pat architektūros tinklas ResNet-101 (lyginant modelius su visomis tikslo funkcijomis).

Įvertinus tikslo funkcijas naudojant Pearsono koreliaciją visiems tinklo modeliams naudojančiams standartinę tripleto tikslo funkciją pavyko pasiekti 100% koreliacijos tikslumą. Vienintelis ResNet-18 tinklo modelis sugebėjo pasiekti 100% Pearsono tikslumą naudodamas kitą tikslo funkciją (su pakeitimais), todėl mokant ne tokius gilius geriau naudoti tripleto tikslo funkciją su pakeitimais. Išanalizavus Pearsono koeficientų reikšmes ResNet architektūros modeliuose

pastebėtas keistas fenomenas. ResNet-18 ir ResNet-101 tinkle vidutinė teigiamos poros koreliacijos reikšmė suvienodėja su neigiamos poros koreliacijos maksimumo reikšme. Norint turėti geresnius rezultatus maksimumo reikšmė turėtų nusileisti žemiau teigiamos poros vidurkio. DLA-34 atvejų taip ir įvyksta. Šitą fenomeną būtų galima tyrinėti sekančiame darbe.

Įvertinus koreliacijos reikšmių persidengimus DLA-34, tinklo modelis apmokytas naudojant standartinę tripleto tikslo funkciją po 10 mokymo epochų, esant Pearsono koreliacijos koeficientui 0.629 net 99.5% teigiamų vaizdų yra identifikuojami kaip panašūs, ir mažiau nei 2% neigiamų vaizdų duoda teigiamą panašumo rezultatą. Kitų tinklų modelių rezultatai taikant koreliacijos koeficiento ribą yra prastesni, todėl galima teikti, kad DLA-34 yra tiksliausias tinklo modelis.

Pasiūlytas modifikuotas neuroninio tinklo CenterNet modelis, pagal apžvelgtas metrikas su sugeneruotu duomenų rinkiniu savo tikslumu pralenkia publikuotame straipsnyje [VMR20] naudojamą gilųjį neuroninį tinklą. Geriausias CenterNet tinklo modelis (DLA-34) lyginant su geriausių VGG16 epochų rezultatais demonstruoja 5% geresnį tikslumą, 8,2% geresnį vidutinio tikslumo rezultatą ir 3,5% geresnį Pearsono koreliacijų tikslumo rezultatą.

Šaltinių sąrašas

- [Are18] Marc-Olivier Arsenault. Lossless Triplet loss.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://towardsdatascience.com/lossless-triplet-loss-7e932f990b24>>
- [BBB+17] Prithaj Banerjee, Ayan Kumar Bhunia, Avirup Bhattacharyya, Partha Pratim Roy, Subrahmanyam Murala. Local Neighborhood Intensity Pattern – A new texture feature descriptor for image retrieval.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://arxiv.org/ftp/arxiv/papers/1709/1709.02463.pdf>>
- [BRP+16] Vassileios Balntas, Edgar Riba, Daniel Ponsa, Krystian Mikolajczyk. Learning local feature descriptors with triplets and shallow convolutional neural networks.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<http://www.bmva.org/bmvc/2016/papers/paper119/paper119.pdf>>
- [CCZ+17] Weihua Chen, Xiaotang Chen, Jianguo Zhang, Kaiqi Huang. Beyond triplet loss: a deep quadruplet network for person re-identification.
[tikrinta 2020-05-31]. Prieiga per internetą:
<http://openaccess.thecvf.com/content_cvpr_2017/papers/Chen_Beyond_Triplet_Loss_CVPR_2017_paper.pdf>
- [DQX+17] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu Yichen Wei. Deformable Convolutional Networks
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://arxiv.org/pdf/1703.06211.pdf>>
- [EY09a] Zhang E., Zhang Y. (2009) Average Precision. In: LIU L., ÖZSU M.T. (eds) Encyclopedia of Database Systems. Springer, Boston, M
[tikrinta 2020-05-31]. Prieiga per internetą:
<https://link.springer.com/referenceworkentry/10.1007%2F978-0-387-39940-9_482>
- [EY09b] Zhang E., Zhang Y. (2009) Precision. In: LIU L., ÖZSU M.T. (eds) Encyclopedia of Database Systems. Springer, Boston, MA
[tikrinta 2020-05-31]. Prieiga per internetą:
<https://link.springer.com/referenceworkentry/10.1007/978-0-387-39940-9_480>
- [EY09c] Zhang E., Zhang Y. (2009) Recall. In: LIU L., ÖZSU M.T. (eds) Encyclopedia of Database Systems. Springer, Boston, MA
[tikrinta 2020-05-31]. Prieiga per internetą:
<https://link.springer.com/referenceworkentry/10.1007/978-0-387-39940-9_479>
- [Gup17] Dishashree Gupta. Transfer learning and the art of using Pre-trained Models in Deep Learning.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://www.analyticsvidhya.com/blog/2017/06/transfer-learning-the-art-of-fine-tuning-a-pre-trained-model/>>
- [HZS+15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun. Deep Residual Learning for Image Recognition.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://arxiv.org/abs/1512.03385>>
- [JM19] Rokas Jurevičius, Virginijus Marcinkevičius. Application of Vision-based Particle Filter and Visual Odometry for UAV Localization.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://www.researchgate.net/publication/333263581>>

- [KZS15] Gregory Koch, Richard Zemel, Ruslan Salakhutdinov. Siamese Neural Networks for One-shot Image Recognition.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://www.cs.cmu.edu/~rsalakhu/papers/oneshot1.pdf>>
- [LD18] Hei Law, Jia Deng. CornerNet: Detecting Objects as Paired Keypoints.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://arxiv.org/pdf/1808.01244.pdf>>
- [LSD15] Jonathan Long, Evan Shelhamer, Trevor Darrell. Fully Convolutional Networks for Semantic Segmentation.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://arxiv.org/pdf/1411.4038.pdf>>
- [Mit17] Vartul Mittal. Top 15 Deep Learning applications that will rule the world in 2018 and beyond.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://medium.com/breathe-publication/top-15-deep-learning-applications-that-will-rule-the-world-in-2018-and-beyond-7c6130c43b01>>
- [NYD+16] Alejandro Newell, Kaiyu Yang, Jia Deng, Ann Arbor. Stacked Hourglass Networks for Human Pose Estimation.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://arxiv.org/pdf/1603.06937.pdf>>
- [Pra18] Selva Prabhakaran. Cosine Similarity.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://www.machinelearningplus.com/nlp/cosine-similarity/>>
- [RMK05] Michael T. Rosenstein, Zvika Marx, Leslie Pack Kaelbling. To Transfer or Not To Transfer.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<http://web.engr.oregonstate.edu/~tgd/publications/rosenstein-marx-kaelbling-dietterich-hnb-nips2005-transfer-workshop.pdf>>
- [SKP15] Florian Schroff, Dmitry Kalenichenko, James Philbin. FaceNet: A Unified Embedding for Face Recognition and Clustering.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://arxiv.org/pdf/1503.03832.pdf>>
- [SLJ+14] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, Andrew Rabinovich. Going deeper with convolutions.
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://arxiv.org/pdf/1409.4842.pdf>>
- [Smi97] Stephen W. Smith "13.Convolution". The Scientist and Engineer's Guide to Digital Signal Processing (1 ed.). California Technical Publishing. ISBN 0-9660176-3-3.
- [TSZ+15] Ben Tan, Yangqiu Song, Erheng Zhong, Qiang Yang. Transitive Transfer Learning
[tikrinta 2020-05-31]. Prieiga per internetą:
<http://csl.t.riit.tsinghua.edu.cn/mediawiki/images/9/97/2015_Transitive_Transfer_Learning.pdf>
- [TSZ+18] Chuanqi Tan, Fuchun Sun, Tao Kong, Wenchang Zhang, Chao Yang, Chunfang Liu. A Survey on Deep Transfer Learning
[tikrinta 2020-05-31]. Prieiga per internetą:
<<https://arxiv.org/pdf/1808.01974.pdf>>

- [Tya19] Deepanshu Tyagi. Introduction to Feature Detection and Matching. [tikrinta 2020-05-31]. Prieiga per internetą: <<https://medium.com/data-breach/introduction-to-feature-detection-and-matching-65e27179885d>>
- [VMR20] Vytautas Valaitis, Virginijus Marcinkevičius and Rokas Jurevicius. Learning Aerial Image Similarity using Triplet Networks. [tikrinta 2020-05-31]. Prieiga per internetą: <<https://www.researchgate.net/publication/339232012>>
- [Wiki0] Kernel (image processing). [tikrinta 2020-05-31]. Prieiga per internetą: <[https://en.wikipedia.org/wiki/Kernel_\(image_processing\)](https://en.wikipedia.org/wiki/Kernel_(image_processing))>
- [YK16] Fisher Yu, Vladlen Koltun. Multi-Scale Context Aggregation by Dilated Convolutions [tikrinta 2020-05-31]. Prieiga per internetą: <<https://arxiv.org/pdf/1511.07122.pdf>>
- [YWS+19] Fisher Yu, Dequan Wang, Evan Shelhamer, Trevor Darrell. Deep Layer Aggregation [tikrinta 2020-05-31]. Prieiga per internetą: <<https://arxiv.org/pdf/1707.06484.pdf>>
- [ZWK19] Xingyi Zhou, Dequan Wang, Philipp Krähenbühl. Objects as Points. [tikrinta 2020-05-31]. Prieiga per internetą: <<https://arxiv.org/pdf/1904.07850.pdf>>
- [ZZK19] Xingyi Zhou, Jiacheng Zhuo, Philipp Krahenb. Bottom-up Object Detection by Grouping Extreme and Center Points. [tikrinta 2020-05-31]. Prieiga per internetą: <<https://arxiv.org/pdf/1901.08043.pdf>>