



Kauno technologijos universitetas

Elektros ir Elektronikos fakultetas

Giliųjų neuroninių tinklų taikymo civilinės infrastruktūros paviršiaus defektų aptikimui iš nuotraukų tyrimas

Baigiamasis magistro projektas

Evaldas Dudonis

Projekto autorius

Doc. Arūnas Lipnickas

Vadovas

Kaunas, 2020



Kauno technologijos universitetas

Elektros ir Elektronikos fakultetas

Giliųjų neuroninių tinklų taikymo civilinės infrastruktūros paviršiaus defektų aptikimui iš nuotraukų tyrimas

Baigiamasis magistro projektas

Valdymo technologijos (6211EX014)

Evaldas Dudonis

Projekto autorius

Doc. Arūnas Lipnickas

Vadovas

Lekt. Vygandas Vaitkus

Recenzentas

Kaunas, 2020



Kauno technologijos universitetas

Elektros ir elektronikos fakultetas

Evaldas Dudonis

Giliųjų neuroninių tinklų taikymo civilinės infrastruktūros paviršiaus defektų aptikimui iš nuotraukų tyrimas

Akademinio sąžiningumo deklaracija

Patvirtinu, kad mano, Evaldo Dudonio, baigiamasis projektas tema „Giliųjų neuroninių tinklų taikymo civilinės infrastruktūros paviršiaus defektų aptikimui iš nuotraukų tyrimas“ yra parašytas visiškai savarankiškai ir visi pateikti duomenys ar tyrimų rezultatai yra teisingi ir gauti sąžiningai. Šiame darbe nei viena dalis nėra plagijuota nuo jokių spausdintinių ar internetinių šaltinių, visos kitų šaltinių tiesioginės ir netiesioginės citatos nurodytos literatūros nuorodose. Įstatymų nenumatytų piniginių sumų už šį darbą niekam nesu mokėjęs.

Aš suprantu, kad išaiškėjus nesąžiningumo faktui, man bus taikomos nuobaudos, remiantis Kauno technologijos universitete galiojančia tvarka.

(vardą ir pavardę įrašyti ranka)

(parašas)

Evaldas Dudonis. Giliųjų neuroninių tinklų taikymo civilinės infrastruktūros paviršiaus defektų aptikimui iš nuotraukų tyrimas. Magistro baigiamasis projektas / vadovas doc. Arūnas Lipnickas; Kauno technologijos universitetas, Elektros ir elektronikos fakultetas.

Studijų kryptis ir sritis (studijų krypčių grupė): Elektronikos inžinerija. Inžinerijos mokslai.

Reikšminiai žodžiai: gilieji neuroniniai tinklai, DNT, paviršiaus defektas, vaizdų analizė.

Kaunas, 2020. 51 p.

Santrauka

Moksliniame darbe nagrinėjami defektoskopijos metodai, plačiau nagrinėjamas vaizdo analizės metodas. Kaip priemonę tikslui įgyvendinti pasitelkiami gilieji neuroniniai tinklai. Darbo tikslas – ištirti giliųjų neuroninių tinklų panaudojimo galimybę civilinės infrastruktūros paviršiaus defektų aptikimui iš nuotraukų.

Metodinėje dalyje aprašoma defektoskopijos nauda pramonei ir infrastruktūrai, vaizdų analizės raida, giliųjų neuroninių ir U-Net tinklų apžvalga. Apžvelgiama tinklų modelių vertinimo įverčiai bei svorinių tinklo koeficientų perkėlimo metodas, duomenų bazė, jos augmentacija ir šio metodo nauda galutiniam tikslui. Darbe apmokomi dirbtiniai neuroniniai tinklai taip, kad pirmu atveju - pateikus nuotrauką tinklas atpažintų tekstūrą, antru – išduotų atsakymą ar nuotraukoje yra defektas ar jo nėra. Trečiu atveju – apmokomas U-Net tipo tinklas taip, kad į tinklo įėjimą padavus paviršiaus nuotrauką, tinklas rastų ir parodytų defektą.

Visi tekstūros atpažinimo tinklai turėjo 30 iteracijų mokymą, po jų buvo gauti tekstūros atpažinimo rezultatai: MobileNet tinklas sugeba atpažinti dalį vaizdų su 68% bendru tinklo tikslumu. NASNet-mobile tinklas pasiekė bendrą 91% tikslumą. InceptionV3 konvoliucinio tinklo tikslumas siekė 95%, įgijo 0,95 F1 įvertį su 0,95 preciziškumu. Antrojo tyrimo metu, tinklai taip pat turėjo 30 iteracijų mokymą ir buvo gauti tokie rezultatai: MobileNet tinklas nuotraukas be sienų paviršiaus defektų skirsto su 87% tikslumu, o nuotraukas su defektais skirsto su 56% tikslumu, nuotraukas be kelio dangos įtrūkimų klasifikuoja su 93% tikslumu, o nuotraukas su įtrūkimais su 17% tikslumu. NASNet-mobile tinklas visiškai negeba klasifikuoti kelio dangos įtrūkimų. Xception tinklas kelio dangos įtrūkimus klasifikuoja su 0,88 F1 ir 0,42 MCC įverčiais. Paskutinio eksperimento metu buvo vykdoma defekto segmentavimas pikselio lygyje, atlikus eksperimentą nustatyta, kad U-Net tinklas yra pajėgus segmentuoti ryškius įtrūkimus, bet nėra pakankamai jautrus aptikti pagrindinio įtrūkimo išsišakojimus. Taip pat, tinklas labai grubią paviršiaus struktūrą priskiria defektui.

Dudonis Evaldas. Research of Deep Neural Networks Application for Surface Defects Detection in Images of Civil Infrastructure. Master's Final Degree Project / supervisor doc. Arūnas Lipnickas; Faculty of Electrical and Electronics Engineering, Kaunas University of Technology.

Study field and area (study field group): Electronics engineering. Engineering science.

Keywords: convolutional neural network, CNN, surface defect, image analysis.

Kaunas, 2020. 51 p.

Summary

In the scientific work defectoscopy methods are analyzed, the image analysis method is analyzed in more detail. Deep neural networks are used as a tool to achieve the goal. The aim of this research is to investigate the possibility of using deep neural networks to detect surface defects in civil infrastructure from images.

The methodological part describes the benefits of defectoscopy for industry and infrastructure, the evolution of image analysis, an overview of deep neural and U-Net networks. The evaluation estimates of network models and the method of transfer learning, database, its augmentation and the benefits of this method were then reviewed to achieve the ultimate goal. Artificial neural networks were trained in such a way that in the first case – when the image is submitted, the network recognizes the texture, in the second case – the network issues an answer whether the image has crack or not. In the third case – a U-Net type network is trained so that after submitting a surface image to the network input, the network finds and displays the defect.

All texture recognition networks had 30 iterations of training, followed by texture recognition results: MobileNet network was able to recognize a part of images with 68% overall network accuracy. NASNet-mobile network achieved an overall accuracy of 91%. InceptionV3 convolutional network accuracy was 95%, gained 0,91 F1 rating with 0,95 accuracy. In the second experiment the networks also had 30 iterations of training and the following results were obtained: MobileNet network allots photos without wall surface defects with 87% accuracy, photos with defects with 56% accuracy, photos without road surface cracks with 93% accuracy and photos with cracks with 17% accuracy. NASNet-mobile network is completely incapable of classifying road pavement cracks. Xception network classifies road pavement cracks with estimates of 0.88 F1 and 0.42 MCC. In the last experiment defect segmentation was performed at pixel level. After the experiment it was found that the U-Net network is capable of segmenting bright cracks but it is not sensitive enough to detect branching of the main crack. Also, the network recognizes a very rough surface structure as a defect.

Turinys

Lentelių sąrašas	7
Paveikslų sąrašas	8
Santrumpų ir terminų sąrašas	9
Įvadas.....	10
1. Metodinė dalis.....	11
1.1. Defektoskopija.....	11
1.2. Vaizdų analizė	12
1.3. Dirbtiniai neuroniniai tinklai	12
1.3.1. Gilusis mokymasis.....	14
1.3.2. Prižiūrimasis mokymas	14
1.4. Konvoliuciniai neuroniniai tinklai.....	15
1.4.1. Konvoliuciniai sluoksniai	16
1.4.2. Mažinimo sluoksniai	16
1.4.3. Pilnai sujungti sluoksniai.....	16
1.5. U-Net.....	17
1.5.1. Architektūra.....	17
1.6. Žinių perkėlimas.....	18
1.7. Modelio vertinimo įverčiai	24
1.7.1. Tikslumas	24
1.7.2. Atkūrimas	25
1.7.3. Taiklumas	25
1.7.4. F_1	25
1.7.5. IoU.....	25
1.7.6. MCC	26
2. Duomenų bazė.....	27
2.1. Augmentacija.....	28
3. Eksperimentinė dalis	30
3.1. Tekstūros atpažinimas	30
3.1.1. VGG16	31
3.1.2. VGG19	31
3.1.3. MobileNet.....	32
3.1.4. NASNet-mobile.....	32
3.1.5. InceptionV3	32
3.1.6. Xception	32
3.1.7. Išvada.....	32
3.2. Defekto atpažinimas	34
3.2.1. VGG16	36
3.2.2. VGG19	36
3.2.3. MobileNet.....	36
3.2.4. NASNet-mobile.....	36
3.2.5. InceptionV3	37
3.2.6. Xception	37
3.2.7. Išvada.....	37
3.3. Defekto segmentavimas pikselio lygyje.....	40
3.3.1. Sienos segmentavimas.....	41
3.3.2. Kelio dangos segmentavimas	43
3.3.3. Šaligatvio segmentavimas	44
3.4. Eksperimentų apibendrinimas	46
Išvados	48
Literatūros sąrašas	49

Lentelių sąrašas

1 lentelė. InceptionV3 tinklo struktūra.....	20
2 lentelė. MobileNet tinklo struktūra	22
3 lentelė. Sumaišymų matrica	24
4 lentelė. SDNET2018 statistinis aprašas.	28
5 lentelė. Tekstūros atpažinimo rezultatai.....	31
6 lentelė. InceptionV3 tinklo paviršiaus tekstūros sumaišymo matrica.....	32
7 lentelė. Sienos defekto atpažinimo rezultatai.....	34
8 lentelė. Kelio dangos defekto atpažinimo rezultatai.	35
9 lentelė. Šaligatvio defekto atpažinimo rezultatai.	35
10 lentelė. Xception tinklo kelio dangos įtrūkimų klasifikavimo sumaišymo matrica.....	37
11 lentelė. Xception tinklo sienų įtrūkimų klasifikavimo sumaišymo matrica.....	38
12 lentelė. MobileNet tinklo šaligatvių įtrūkimų klasifikavimo sumaišymo matrica.....	39

Paveikslų sąrašas

1 pav. Vienasluoksniai ir daugiasluoksniai tinklai.....	13
2 pav. Dirbtinių neuroninių tinklų struktūros: a) be grįžtamojo ryšio; b) su grįžtamoju ryšiu;.....	14
3 pav. Modifikuotas konvoliucinis neuroninis tinklas	16
4 pav. U-Net tinklo architektūra.....	18
5 pav. VGG-16 tinklo architektūra	19
6 pav. VGG-19 tinklo architektūra	20
7 pav. InceptionV3 tinklo architektūra.....	21
8 pav. NASNet struktūra	23
9 pav. Xception tinklo architektūra.....	24
10 pav. SDNET2018 duomenų bazė sudaryta iš (a)-smulkių įtrūkimų, (b)-gilių trūkių, (c)-šešėlių, (d)-dėmių, (e)-įtrūkimų grubiame paviršiuje, (f)-tuštumos, (g)-briaunų, (h)- jungčių ir iškorėjusių paviršių, (i)-atsitiktinių fono kliūčių;	27
11 pav. a – originalus vaizdas, b – apverstas, c – pasuktas 90°, d – pasuktas 180°, e – pasuktas 270°, f – apverstas originalus ir pasuktas 90°, g - apverstas originalus ir pasuktas 180°, h – apverstas originalus ir pasuktas 270°.....	29
12 pav. InceptionV3 suklasifikuotų nuotraukų pavyzdys	33
13 pav. Xception tinklo suklasifikuotų kelio dangos nuotraukų pavyzdys.....	38
14 pav. Xception tinklo suklasifikuotų sienų nuotraukų pavyzdys	39
15 pav. MobileNet tinklo suklasifikuotų šaligatvio nuotraukų pavyzdys.....	40
16 pav. „CrackForest“ duomenų bazės paveikslėliai (viršuje) ir individualūs savybių žemėlapiai (apačioje).....	41
17 pav. Sienos paviršiaus segmentavimo rezultatai, kai sienoje yra defektas	42
18 pav. Sienos paviršiaus segmentavimo rezultatai, kai sienoje nėra defekto.....	43
19 pav. Kelio dangos paviršiaus segmentavimo rezultatai, kai dangoje yra defektas	43
20 pav. Kelio dangos paviršiaus segmentavimo rezultatai, kai dangoje nėra defekto.....	44
21 pav. Šaligatvio paviršiaus segmentavimo rezultatai, kai paviršiuje yra defektas	45
22 pav. Šaligatvio paviršiaus segmentavimo rezultatai, kai paviršiuje nėra defekto.....	46
23 pav. Sisteminis algoritmas (WDP – tekstūros atpažinimas; W – sienos; D – kelio danga; P – šaligatvis; CD – atpažintas defektas).....	47

Santrumpų ir terminų sąrašas

Santrumpos:

RGB – spalvų maišymo sistema.

CNN – konvoliucinis neuroninis tinklas (angl. *convolutional neural network*).

F_1 – tikslumo matas.

IoU – tikslumo matas, dar vadinamas Džakardo indeksu (angl. Jaccard index).

MCC – tikslumo matas.

FC – pilnai sujungtas sluoksnis (angl. *fully connected layer*).

CD – aptiktas defektas (angl. *crack detected*)

Įvadas

Dabartinėje visuomenėje vis aktualesnėmis tampa vaizdo atpažinimo ir analizavimo sistemos. Išplitus vaizdo apdorojimo sistemoms, kilo poreikis ne tik gerinti vaizdo kokybę, bet ir atlikti skaitmeninių vaizdų analizę, atpažinti objektus, objekto kontūrus, segmentuoti objektus ir panašiai. Todėl ši sritis stipriai išsiplėtė visomis kryptimis ir yra naudojama medicinai, archeologijai, pramonei, kriminalistikai, transporte ir t.t.

Defektoskopija – bendras daugelio metodų pavadinimas, paprasta akimi nematomiems gaminių defektams rasti (magnetinis, vaizdinis, ultragarsinis, kapiliarinis, rentgeno, ir t.t.). Šie metodai, dažnai vadinami neardomais tyrimo metodais. Įtrūkimai ir kiti defektai: tuštumos, korėti paviršiai, šlako junginiai ir t.t. dažniausiai randami vizualiniu, kapiliariniu ir magnetiniu metodais. Detalių matmenys gali būti tikrinamos pneumatiniu metodu arba universaliais matavimo įrankiais. Paviršiniams įtrūkimams nustatyti, tinka ultragarsiniai, indukciniai ir akustiniai metodai. Taigi, vieno ar kito metodo parinkimas defektams surasti priklauso nuo defekto vietos, detalės medžiagos, jos dydžio, konfigūracijos ar kitų faktorių [1].

Per ilgą laiką, žmogus suvokia, įgyja patirties atpažinti tam tikrus objektus. Tai atrodo natūralus, pastovus procesas, bet tik ne vaizdo analizavimo sistemoms. Sukurti dirbtinių neuroninių tinklų metodai puikiai nustato ir atpažįsta objektus ar kontūrus. Šiai grupei priskiriami ląstelių, konstrukcijų ar kiti defektai. Šiuo metu žmonės patys identifikuoja paviršiaus pažeidimus, sprendžia apie jos būklę ir remonto darbus, todėl plečiantis infrastruktūrai ir pramonei tampa aktualu ieškoti automatizuoto sprendimo, sumažinančio žmogiškųjų išteklių naudojimą. Skaitmeniniame vaizde sukaupiama daug daugiau informacijos nei būtų galima įvertinti tyrinėjant žmogaus akimi [2].

Dirbtinis intelektas – tai viena iš šiuo metu aktualiausių temų technologijų ir inovacijų srityje. Dirbtinio neuroninio tinklo sistemos geba atlikti užduotis, susijusias su žmogaus protu, tokias kaip vaizdų suvokimas bei sprendimų priėmimas. Skaičiavimais ir giliu mokymusi pagrįsti, patys save apmokantys algoritmai leidžia analizuoti milžiniškus duomenų kiekius, atpažinti, klasifikuoti bei pateikti išvadas apie analizuotą objektą.

Tiriamąo darbo tikslas

Ištirti giliųjų neuroninių tinklų panaudojimo galimybę civilinės infrastruktūros paviršiaus defektų aptikimui iš nuotraukų.

Tiriamąo darbo uždaviniai

- Apžvelgti vaizdų analizavimo metodus, jų plėtrą ir panaudojimo galimybes.
- Apžvelgti dirbtinius neuroninius tinklus, jų veikimą, struktūrą ir tipus.
- Parinkti duomenų bazę ir ją pritaikyti tiriamajam darbui.
- Ištirti žinių perkėlimo panaudojimo galimybę civilinės infrastruktūros defektų aptikimui iš nuotraukų.
- Apmokyti giliuosius neuroninius tinklus su pritaikyta duomenų baze, atpažinti infrastruktūros defektus.
- Apžvelgti U-Net tinklo architektūrą, jo veikimą ir galimybes.
- Pritaikyti U-Net tinklą civilinės infrastruktūros defektų segmentavimui pikselio lygmenyje.

1. Metodinė dalis

1.1. Defektoskopija

Defektas – tai medžiagų, konstrukcijų ar dirbinių kokybės parametrų rodiklių nuokrypiai nuo normų reikalavimų, atsirandantys transportuojant, eksploatuojant ir jas gaminant. Po antro pasaulinio karo prasidėjusi technologijų plėtra paskatino naudoti neardančios kontrolės testavimo metodus, kurie iki šių dienų tobulinami ir naudojami defektoskopijoje. Defektoskopija– medžiagų kokybės kontrolė jų nesuardant. Defektoskopiją galima apibūdinti kaip gaminių defektų radimo technologinių metodų visumą.[4].

Defektai pagal pavojingumo laipsnį gali būti: pavojingi, mažai pavojingi, kritiniai; pagal suradimo būdą: paslėpti (vidiniai), aiškūs (išoriniai); pagal pašalinimo galimybę: pašalinami ar nepašalinami. Gamybos defektai gali atsirasti lydymo ir liejimo metu, kalant, valcuojant, sujungiant litavimo, suvirinimo, klijavimo būdais bei terminio, mechaninio ir elektrocheminio apdirbimo metu. Eksploataciniai defektai atsiranda gaminiui išdirbus tam tikrą laiką ir yra skirstomi į du pagrindinius pogrupius: pažeidimai ir plyšiai dėl korozijos (tarpkristaliniai, vientisi, taškiniai žaizdų ir įpjovų pavidalo, išsisluoksniuojantys ir kt.), įvairios prigimties (terminiai, vienkartiniai apkrovų nuovargio, korozinio nuovargio, terminio nuovargio). Pavojingiausias ir labiausiai paplitęs detalių defektas yra nuovargio, ar kitaip, nusidėvėjimo plyšiai. Jie atsiranda dėl įvairių defektų įtakos, ne tik dėl kintamų apkrovų poveikio [1].

Paviršiniams defektams nustatyti yra atliekami bandymai skverbikliais. Šis metodas dar vadinamas kapiliarine defektoskopija. Tai yra paviršiaus tyrimo metodas, kuriuo galima tirti visas neporingas medžiagas. Skysčiu – „skverbikliu“ yra padengiamas medžiagos paviršius. Skverbiklis prasiskverbia į defekto (pažeidimo) vidų. Kita medžiaga – ryškalas – parodo, kurioje vietoje skverbiklis prasiskverbė į pažeidimo vidų. Galima iširti visas neporingas medžiagas, naujų arba eksploatuojamų konstrukcijų, atramų, suvirinimo siūlių, kėlimo kėlpų, tvirtinimo priemonių ir kitus paviršius. Šiuo metodu nustatomos tuštumos, įtrūkimai ir kiti atviri paviršiniai defektai [1].

Neardančios kontrolės metodai analizuojamame objekte padeda nustatyti defektus jo nesugadindami, t.y. nesulydydami, negręždami, nepjaustydami ar nesuardydami detalėmis ir atlieka tyrimus realiu laiku. Tokie metodai priklauso optinės inžinerijos kryptčiai. Šio tipo metodų yra įvairių: tai gali būti fototamprumo efektu pagrįsti optiniai metodai, muaro juostomis pagrįsti matavimo metodai, ar vidurkinta laike lazerinė holografija. Kiekvienu atveju susidarantis interferencinių juostų žemėlapis priklauso nuo optinės sistemos ypatumų, o ne tik nuo judesio, matuojamosios sistemos būsenos, svyravimo ar statinių deformacijų lauko. Atspindžio muaro interferencinių juostų laukas priklauso nuo apšviestumo charakteristikų, statinės gardėlės tankio. Fototamprumo atveju priklausys nuo šviesos poliarizacijos, nuo stebėjimo kampo. Jei nagrinėti vidurkintą laike lazerinę holografiją, tai interferencinių juostų laukas priklausys nuo apšvietimo kampo, lazerio bangos ilgio ir stebėjimo kampo [5].

Šiais laikais, atsiradus galimybėms, išaugus skaičiavimo resursams, vaizdo kokybę pagerinus iki realybės prilygstančios, galima šiai medžiagų ar paviršiaus kontrolei priskirti ir dirbtiniais neuroniniais tinklais paremtas vaizdo analizavimo sistemas. Tai tema, kuri apima ne vieną dešimtmetį tyrimų, išradimų ir atradimų. Tobulėjant skaičiavimo technikai atsiveria dar nepatirtos galimybės, gerokai atpiginusios šią ganėtinai brangią optinės inžinerijos sritį. Eksperimentai persikelia iš laboratorijų į kompiuterių ekranus [6].

1.2. Vaizdų analizė

Šiuolaikinės sistemos pagrįstos optikos principais leidžia kokybiškai užfiksuoti ir atkurti tiriamojo objekto vaizdą skaitmeninėje formoje ir atlikti tam tikrą jo analizę. Vaizdų analizę sudaro vaizde esančių komponentų klasifikavimas pagal nustatytus požymius. Vaizdų apdorojimo operacijas galima klasifikuoti įvairiais būdais. Norint gauti geresnį vaizdą, lyginant su pradiniu, siekiama manipuliuoti vaizdu, taip išskiriant dominančius regionus. Objektivią vaizdų analizę sudaro labai didelis duomenų kiekis, todėl vaizdų analizės užduotis, sumažinti nereikšmingos informacijos kiekius, nustatyti mažesnių žingsnių seką, skirtą tiriamai problemai spręsti ir kiekvieno etapo pabaigoje pagerinti vaizdą. Baigiamuoju etapu atliekamas vaizdo požymių išskyrimas iš aplinkos. Svarbu žinoti, kad vaizdų analizė susideda iš: vaizdų paruošimo, vaizdų filtravimo, vaizdų pakeitimo ir suskirstymo, požymių išskyrimo ir charakteristikų analizės.

Vaizdų analizės metodais apjungiamas mokslas ir šiuolaikinės technologijos. Skaitmeniniame vaizde sukaupiama daug daugiau informacijos nei būtų galima įvertinti pro įprastą mikroskopą tyrinėjant žmogaus akimi, skaitmeninės analizės algoritmai suteikia reikiamas priemones uždaviniams spręsti, leidžia labai tiksliai nusakyti molekulinę ligos žymenų pasiskirstymą audiniuose, patologinių pokyčių laipsnį, ląstelių tarpusavio sąveiką. Skaitmeninis signalas pagreitina biožymenų nustatymą, leidžia atlikti kiekybinius jų matavimus ir naudoti duomenis mokslo ir kokybės užtikrinimo tikslais.

Eksperimentuose pritaikant vaizdų analizės metodą labai svarbu, kad skaitmeniniuose vaizduose užfiksuotas objektas būtų neiškraipytas, visame vaizdo plote esantys elementai būtų gero kontrasto, sodrumo, detalumo ir ryškumo. Gautų rezultatų kokybę nusako apšvietimo sąlygos, atstumas tarp objekto ir fotoaparato, tiriamojo objekto paruošimas, optinis vaizdo iškraipymas. Pagal B.K. Behera, tinkamas tiriamojo objekto apšvietimas yra vienas iš svarbiausių veiksnių, darantis įtaką atkuriamiems skaitmeniniams vaizdams ir jų analizavimo kokybei. Dažniausiai aplinkos šviesa eksperimentuose yra nepakankama, gali atsirasti optiniai iškraipymai dėl optinių jutiklių, aplinkos įtakos ir objektų paviršiaus netolygaus apšviestumo. Todėl yra parenkama apšvietimo sistema pagal sprendžiamą problemą ir naudojamas dirbtinis objekto apšvietimas. Kad nenukentėtų vaizdo kokybė, neišsikreiptų objekto formos ir atsiskleistų tiriamos objekto savybės [8].

Vaizdų analizės metodai plėtojami skirtingose mokslo bei pramonės srityse: medicinoje, karinėje pramonėje, robotikoje, dokumentų apdorojime ir kt. Technologijoms sparčiai tobulėjant, pritaikius vaizdų analizavimo technikas, padidinamas matavimų tikslumas, sumažinami tyrimo kaštai ir pagreitinami tyrimo procesai. Vaizdų tyrimas taikomas ne tik makro-lygmeniu, bet tampa labai aktualus mikroskopijoje. Vaizdų analizavimo sistemos leidžia iš vaizdų išskirti dominančią informaciją ir sumažinti analizuojamų duomenų kiekį, palengvindamas ir pagreitindamas matematinės manipuliacijas su vaizdais. Skaitmeniniai vaizdų analizės metodai, yra informatyvūs, gaunamus rezultatus yra paprasta analizuoti, o patį procesą – automatizuoti [7].

Behera B.K., teigia, kad vaizdų analizės metodai reikalauja vaizdo susidarymo principų išmanymo, optikos teorijos žinių. Tai padeda teisingai nustatyti pradines eksperimento sąlygas ir gauti informatyvius ir patikimus tyrimų rezultatus [9].

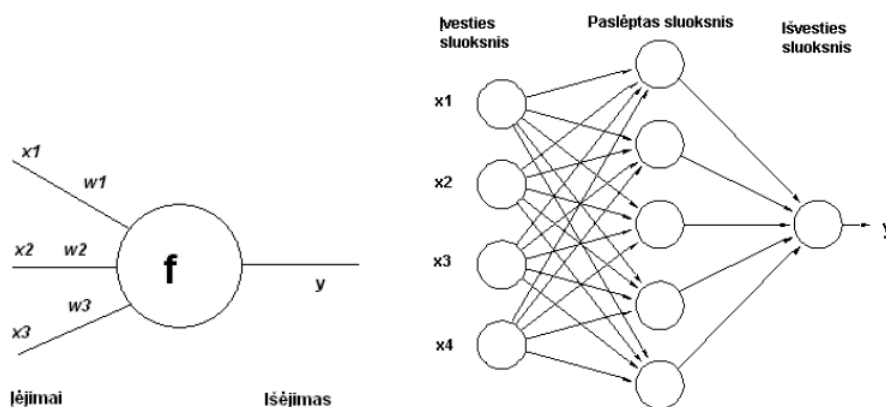
1.3. Dirbtiniai neuroniniai tinklai

Dirbtiniai neuroniniai tinklai buvo pradėti nagrinėti XX a. šeštajame dešimtmetyje. Nepaisant to, kad dirbtinis intelektas pradėtas tyrinėti, bei kurti, daugiau nei prieš 50 metų, tačiau, per pastarąjį dešimtmetį stipriai išaugus skaičiuojamiesiems resursams, tai plačiai atvėrė duris skaitmeniniam

intelektui. Dirbtinių neuroninių tinklų plėtra pačioje pradžioje buvo motyvuojamas tuo, kad žmogaus smegenys skaičiuoja ir priima sprendimus visai kitokiu būdu nei kompiuteris. Remiantis šia prielaida, buvo sukurta informacijos apdorojimo struktūra, gebanti netiksliai imituoti kai kuriuos informacijos apdorojimo procesus, vykstančius gyvųjų organizmų smegenyse [10].

Tai atvėrė duris naujos kartos programavimui – mašininiam mokymuisi. Klasikiniame programavime, žmogus turi nustatyti tam tikras taisykles, kuriomis vadovaujantis, pateikus kažkokius duomenis programa galės priimti sprendimus. Tuo metu, čia yra pateikiami duomenys bei žinomi atsakymai, o išėjime gaunamos taisyklės. Kuriais toliau pasinaudojant, galima gauti originalius atsakymus. Visą tai sėkmingai patvirtino britų logikas bei matematikas Alanas Tiuringas pristatęs *Tiuringo* testą, kurio tikslas yra dirbtinio intelekto patikrinimas. Testas laikomas išlaikytu, jeigu „tardytojas“ kompiuterį palaiko žmogumi.

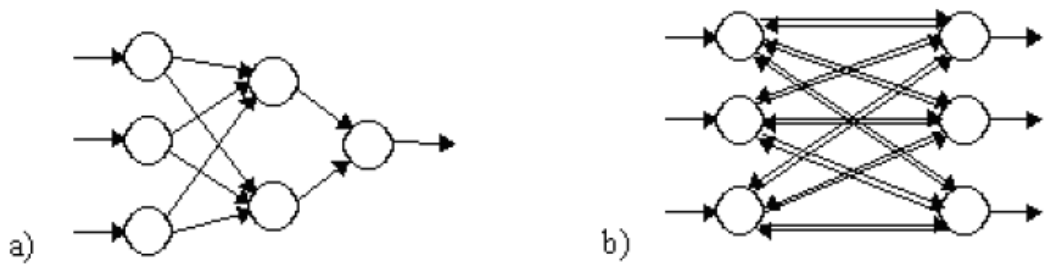
Dirbtinių neuroninių tinklų struktūra gali būti vieno sluoksnio arba daugiasluoksnė. Tiek viena, tiek kita struktūra gali turėti daugiau nei vieną įėjimą bei išėjimą (žr. 1 pav.). Skirtumas tarp jų yra tas, kad daugiasluoksnių perceptronų tinklas turi vieną ir daugiau paslėptų sluoksnių, padedančių spręsti sudėtingas problemas [10].



1 pav. Vienasluoksniai ir daugiasluoksniai tinklai

Šie tinklai būna įvairių struktūrų, kurios skiriasi grįžtamojo ryšio technologija, paslėptų sluoksnių skaičiumi ar kt. Dažniausiai nustatomos šios struktūros:

1. Be grįžtamojo ryšio (angl. *Feedforward*) tinklai – tai tokie tinklai, kuriuose informacija keliauja iš įėjimo sluoksnio paslėptųjų sluoksnių link, ir nuo jų išėjimo sluoksnio link (žr. 2 pav. a.).
2. Su grįžtamoju ryšiu (angl. *Feedback*) tinklai – tokios struktūros tinkluose informacija gali keliauti ne tik viena kryptimi link išėjimo, bet gali ir grįžti iš vieno paslėpto sluoksnio į kitą, iš jų į įėjimą ir panašiai (žr. 2 pav. b.).
3. Be pastarųjų dviejų, egzistuoja aibė įvairiausių tinklo struktūrų.



2 pav. Dirbtinių neuroninių tinklų struktūros: a) be grįžtamojo ryšio; b) su grįžtamoju ryšiu;

1.3.1. Gilusis mokymasis

Pagrindinis dirbtiniam intelektui keliamas iššūkis yra užduočių sprendimas, kurias yra paprasta apdoroti žmogui, tačiau sudėtinga suformuluoti kompiuteriui, problemas, kurias žmonės sprendžia intuityviai, kas atrodo savaime suprantama ir įprasta, tokias kaip veidų ar daiktų atpažinimas. Pagrindinis tokios problemos sprendimas yra paruošti kompiuterius, mokytis iš savo patirties, bei naudojantis koncepcijų hierarchija suprasti aplinką ir pasaulį, kiekvieną koncepciją susiejus su paprastesnėmis koncepcijomis [11].

Dirbtinis neuroninis tinklas, turintis tris ar daugiau nelinejinius paslėptus sluoksnius, gali būti vadinamas giliu neuroniniu tinklu. Tokio tipo algoritmus pastaruoju metu imta vadinti giliojo mokymo algoritmais (angl. *deep learning*). Tam, kad būtų galima neuroninį tinklą vadinti giliuoju, jis turi atitikti keletą pagrindinių sąlygų:

1. Turi būti galimybė praplėtimui – tinkle turi būti galima įterpti naujų sluoksnių;
2. Visi kintami tinklo parametrai turi būti išmokstami, kad nereikėtų tų pačių problemų spręsti kelis kartus.

Gilusis mokymasis leido skaičiavimo modeliams, susidedantiems iš kelių apdorojimo sluoksnių, mokytis iš pateiktų duomenų bei pateikti prognozes jiems. Šie metodai labai pagerino naujausius vizualinių objektų atpažinimo, kalbos atpažinimo, objektų aptikimo ar grupavimo bei daugelio kitų sričių raidą. Ši mokymosi technologija neatsiejama dalis nuo šiuolaikinės visuomenės, ji personalizuoja žiniatinklio paieškas, filtruoja turinį socialiniuose tinkluose, pagal tai kas mums aktualu, remiantis ir kitais duomenimis, paruošia asmenines rekomendacijas elektroninėse parduotuvėse. O ką jau kalbėti apie tokias grandiozines sistemas kaip „Siri“, „Cortana“, „Bixby“, „Alexa“, visos jos paremtos ta pačia giliojo mokymosi (angl. *deep learning*) technologija [12].

Mokymasis iš patirties, suformulavus tam tikras taisykles, nėra reikalingas papildomas žmogaus įsikišimas sprendžiant problemas. Koncepcijų hierarchija leidžia kompiuterį išmokyti sudėtingu mechanizmu, pradedant kurti nuo pačių paprasčiausių

1.3.2. Prižiūrimasis mokymas

Dažniausiai naudojama mašininio mokymosi forma, nepriklausomai ar tai gilusis ar ne, yra prižiūrimasis mokymasis (angl. *supervised learning*). Šiai mokymosi sričiai yra naudojamos didelės apimties informacijos grupės (angl. *data sets*), tarkim – vaizdai, tam kad būtų galima juos panaudoti kaip mokomąją medžiagą atpažinti tam tikrus norimus objektus, pavyzdžiui, automobilius, kelio ženklus, žmones, nejudančius objektus, o galbūt ir mažesnius, tuos pačius kelio įtūkimus ar

mikroskopinio lygio ląstelių pokyčius. Visoms kategorijoms yra priskiriamos žymės (angl. labels), tam, kad vėliau būtų galima atskirti, kas yra kas.

Mokymo metu kompiuteriui, kaip įvesties duomenys yra pateikiami vaizdai, išėjime gautas mokymo rezultatas yra vektorinės formos, jame gaunami sutapimo įvertinimo su kiekviena kategorija rezultatai. Tikslas yra gauti kuo geresnį atitikimą kiekvienai kategorijai, tačiau to gauti prieš mokymą negalima tikėtis. Toliau apskaičiuojama paklaidos tikimybės funkcija bei atstumas tarp gauto vektoriaus ir užsibrėžto, pagal kurį neuroniniame tinkle yra modifikuojami parametrai, vadinamieji „svoriai“, tam, kad būtų sumažinta paklaidos tikimybė identifikuojant objektą kito žingsnio metu [13]. „Svoriai“ tai sveikieji skaičiai, su kuriais kaip su rankenėlėmis galima reguliuoti įėjimo – išėjimo funkciją. Šių parametrų įprastoje gilaus mokymosi sistemoje gali būti šimtai milijonų, kaip ir vaizdų ar kitos informacijos naudojamos sistemos mokymui.

Praktikoje yra naudojami įvairiausi metodai paklaidos tikimybei sumažinti. Vienas iš jų vadinamas stochastinio gradiento aproksimacija (angl. *stochastic gradient descent*), kurio metu kiekvienai mokymo iteracijai yra imamas atsitiktinis pavyzdys ir modifikuojami svoriai, iki paklaidos tikimybė nustoja mažėti. Šiuo metodu pasiekiamas norimas tikslumas, nepanaudojus visų mokymui skirtų duomenų, kai jų yra labai daug. Be šio metodo yra ir kitų aproksimacijos metodų, tokių kaip gradiento aproksimacijos bei antros eilės gradiento aproksimacijos metodai, tačiau stochastinis aproksimacijos metodas užsibrėžtą tikslumą pasiekia greičiau, nei minėtieji [12].

Galiausiai mokymo pabaigoje sistema yra išbandoma, kaip įvesties duomenys jei pateikiamos duomenų grupės, kurios nebuvo naudojamos mokymo procese. Šio proceso metu yra išsiaiškinama neuroninio tinklo darbo greitis bei tikslumas.

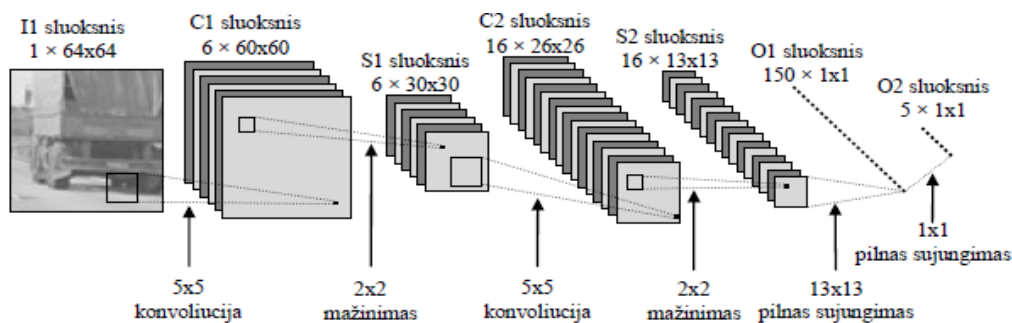
1.4. Konvoliuciniai neuroniniai tinklai

Sprendžiant labiau komplikuotas problemas, tokias kaip vaizdų apdorojimo ir jų atpažinimo ar klasifikavimo problemas yra naudojami daugiasluoksniai neuroniniai tinklai – konvoliuciniai tinklai. Gilieji konvoliuciniai tinklai sukėlė proveržį kai išaugo grafinių procesorių našumas. Nuolatos yra bandoma pagerinti mašininio mokymosi metodus, stengiamasi panaudoti kuo didesnes duomenų imtis ir pasirinkti tinkamesnes technikas, tam, kad apsisaugoti nuo duomenų įsiminimo (angl. *overfitting*) [12]. Apmokant tinklą galima pasiekti didelį tikslumą, tačiau patikros metu atliekant sekančias iteracijas jis išlieka žemas, kas rodo, jog tinklas yra permokintas. Tai reiškia, kad jis mokosi tokius objekto požymius, kurie trukdo tinklo augimui, klasifikavimui ar atpažinimui. Ši problema yra sprendžiama keliais etapais, pirmo etapo metu yra stabdomas apmokymo procesas, kai neuroninio tinklo klasifikavimo tikslumas pradeda blogėti, o antrojo etapo metu yra taikoma nuobaudos neuroninio tinklo svoriams ir išmetimo operacijos.

Konvoliuciniai tinklai gali turėti trijų tipų sluoksnius:

1. Konvoliucinius;
2. Mažinimo;
3. Pilnai sujungtus.

Šie sluoksniai yra išdėlioti taip, kad ankstesnio sluoksnio rezultatas būtų sekančio sluoksnio įėjimas ir t.t. (žr. 3 pav.).



3 pav. Modifikuotas konvoliucinis neuroninis tinklas

1.4.1. Konvoliuciniai sluoksniai

Pagrindinė konvoliucinio neuroninio tinklo dalis yra konvoliuciniai sluoksniai. Kiekvienas iš jų turi po kelias dvimates plokštumas su neuronais, kurios vadinamos – požymių žemėlapiais (angl. *feature map*). Konvoliucinio sluoksnio parametrai yra sudaryti iš mokomųjų filtrų rinkinio. Kiekvienas šiame sluoksnyje esantis neuronas yra sujungtas su grupele neuronų iš prieš tai buvusio sluoksnio ir sudaro jautrumo lauką, kuriame esantys neuronai, iš to pačio požymių žemėlapio, dalinasi tais pačiais svoriais. Požymių žemėlapiai gali būti sujungti pilnai arba dalinai tarp šalia esančių sluoksnių, tai gali būti apibrėžta iš anksto, arba generuojama atskirai.

Atliekant skaičiavimus, pradedama yra nuo konvoliucijų tarp kiekvieno iš įeities požymių žemėlapio ir jam atitinkančio neurono sumuojant įeities reikšmes padaugintas iš atitinkamų svorių. Prie gauto rezultato yra pridėjama pastovi vertė, kuri būna formuluojama neuroninio tinklo mokymo procese, ir galutinis rezultatas paduodamas į aktyvavimo funkciją [14].

Šio sluoksnio rezultatas – požymių žemėlapių rinkinys, kurių dėka tinklas mokomas atpažinti detales ar požymius tokius kaip briaunos ar regionai kuriuose dominuoja kažkokios savybės.

1.4.2. Mažinimo sluoksniai

Požymių žemėlapių suspaudimui yra naudojami mažinimo – telkimo sluoksniai (angl. *pooling layers*). Šie dažniausiai yra naudojami tarp dviejų konvoliucinių sluoksnių, kad būtų galima prieš tai buvusiam sluoksnyje esančius požymių žemėlapio neuronus apjungti į vieną neuroną kurį galima būtų perkelti į kitą sluoksnį, kaip į įėjimą. Telkimo operacija progresyviai sumažina reprezentuojamos informacijos dydį, o mažesnio kiekio informacijos yra išgaunamas didesnis našumas, taip pat, mažiau informacijos reiškia mažiau parametrų ir tai sumažina tikimybę tinklo permokymui. Telkimo operaciją galima naudoti tik tada, kai siekiama sumažinti architektūros jautrumą įvairiems veiksniams. Šis sluoksnis turi lygiai tiek pat požymių žemėlapių kaip ir prieš tai buvęs sluoksnis, ir kiekvienas iš jų yra jungiamas tik su jam atitinkančiu žemėlapio 1×1 jungimu. Šių sluoksnių naudojimas padidina neuroninio tinklo statistinį efektyvumą, nes po konvoliucijos operacijų yra pasiekiami ir išmokstami nekintantys požymiai, todėl praktikoje plačiausiai naudojama maksimalaus telkimo operacijos [15].

1.4.3. Pilnai sujungti sluoksniai

Pilnai sujungtas sluoksnis (angl. *fully connected layers*) labai panašus į konvoliucinius sluoksnius tik yra naudojamas atpažinimui bei klasifikavimui, todėl jie privalo būti naudojami, kad būtų galima pasiekti tikslą. Vienintelis jų tarpusavio skirtumas yra, kad neuronai konvoliuciniuose sluoksniuose yra sujungti tik su lokaliais regionais. Pilnai sujungtų sluoksnių gali būti vienas ar keli, tačiau jie visada bus po mažinimo ir konvoliucinių sluoksnių. Šiame sluoksnyje neuronai sujungti su visais

neuronais iš prieš tai buvusio sluoksnio, jų yra tiek, kiek kategorijų norima atpažinti su šiuo tinklu [16;17].

1.5. U-Net

Didžiausia įprastų dirbtinių neuroninių tinklų problema, yra tai, kad tam reikalinga daug mokymo duomenų, kad būtų pasiekti aukšti kokybės rezultatai.

Kita ne mažiau svarbi problema susijusi su kodavimo – dekodavimo vieta, yra ta, kad matmenų keitimas kodavimo sluoksniuose praranda smulkius vaizdo elementus, kurie gali būti svarbūs klasifikavimui ir kurie negali būti atkurti dekodavimo sluoksnių,

Architektūrų tokių kaip U-Net kūrimu yra bandoma išspręsti tinklo mokymo problemas, kai yra stingama duomenų. Šio tinklo tikslas - naudojant gerokai mažesnius duomenų rinkinius tinklo mokymui, apmokyti tinklą nedarant neigiamos įtakos tinklo įverčiams. Tai atveria naujas galimybes konvoliuciniams tinklams, spręsti užduotis, kurių iki šiol nebuvo galima įveikti, dėl duomenų stygiaus. Taip pat, ši architektūra, naudoja kiekvieną kodavimo sluoksnio išvestį, atitinkamame dekoderio sluoksnyje, tai padeda atskirti net menkiausias detales. Visa tai, U-Net architektūrą padaro tinkamą trūkių nustatymui [18].

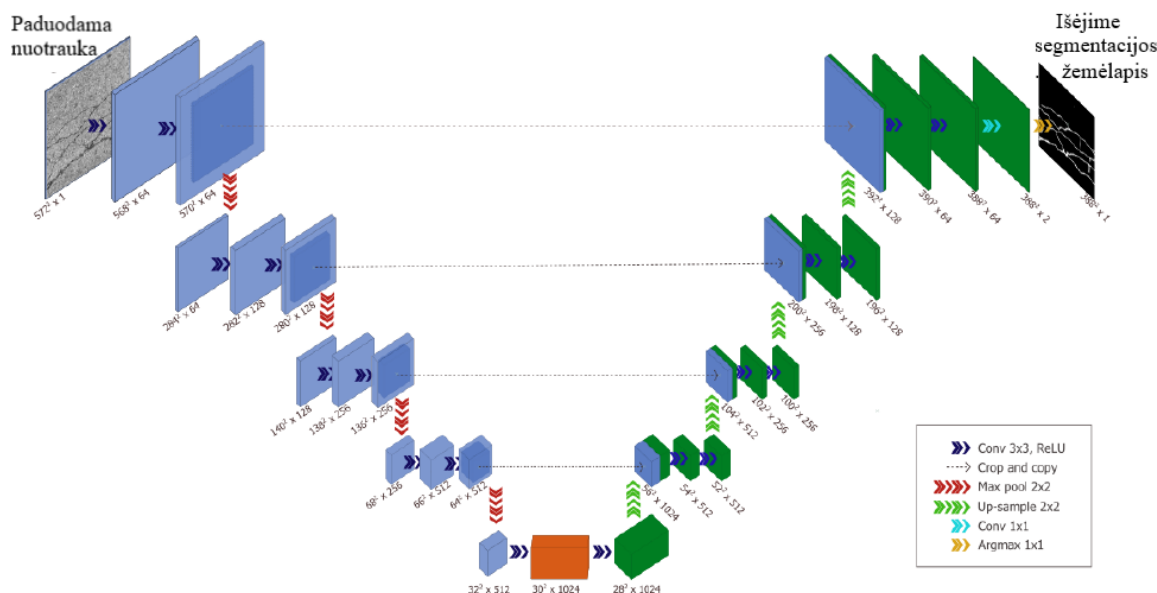
1.5.1. Architektūra

Tinklas sukurtas labiau biomediciniųjų vaizdų, tokių kaip ląstelių ar bakterijų, venų bei kapiliarų segmentavimui. Kad ir kaip atrodytų, tačiau biomediciniųjų vaizdų ir kelio trūkių vaizdų segmentavimas turi daug bendro. Kad ir tai, jog užduotys yra ribojamos turimos tinklo mokymo medžiagos kiekiu, abi jos sudarytos iš smulkių vaizdo detalių, kurios gali būti prarastos suspaudimo (angl. *down-sampling*) metu, ir abi reikalauja kuo tikslesnės segmentacijos išėjimo. Todėl dėl panašių įėjimo duomenų bei siekiamų rezultatų, U-Net yra visiškai tinkamas ir kelio trūkių segmentavimui.

U-Net yra pilnai konvoliucinis tinklas, jame nėra pilnai sujungtų sluoksnių. Visas neuroninis tinklas gali būti išskaidytas į dvi unikalias kodavimo – dekodavimo dalis, kurios forma ir atspindi U-Net pavadinimą (žr. 4 pav.).

Pirmoji – kodavimo dalis yra įprastas konvoliucinis neuroninis tinklas, kuriame vaizdinė informacija yra suspaudžiama, o požymių informacija išplečiama. Tai atliekama įvesties vaizdą perleidžiant per seką konvoliucijų, bei siekiant duomenų normalizacijos panaudojama ReLu (angl. *Rectified Linear Unit*) aktyvacijos funkcija, ir galiausiai panaudojama maksimalaus telkimo operacijos.

Antroji – dekodavimo dalis. Segmentavimo užduočiai dekodavimo sekcija skirta įvertinti požymių vektorius, kad būtų galima sudaryti tikimybių pasiskirstymą kiekvienam paduodamo vaizdo pikseliui atskirai, o ne visam atvaizdui. Tai apjungimų ir transponuotų konvoliucijų sistema, turinti lygiagrečią prieigą prie atitinkamų kodavimo tinklo dalies sluoksnių, kad neprarasti smulkių vaizdo elementų požymių [18].



4 pav. U-Net tinklo architektūra

Standartiškai, dekodavimo sluoksnis yra veikiamas tik kodavimo sluoksnio struktūros, tačiau turi būti pakankamas išplėtimas, kad rezultate gautume lygiaverčio pločio ir aukščio požymių žemėlapi, lyginant su paduodamu vaizdu įėjime. Taip pat, visos smulkios vaizdo detalės, kurios nėra užfiksuotos žemiausio matmens vektoriaus, įprastame tinkle būna prarastos. U-Net architektūra stengdamasi to išvengti įgalina prieigą kiekvienam iškodavimo sluoksniui prie atitinkamo kodavimo sluoksnio. Paskutinis iškodavimo sluoksnis, turi prieigą prie pirmojo kodavimo sluoksnio dar prieš įvykstant konvoliucijoms. Naudojant šį metodą yra padedama iškodavimo sluoksniams atkurti bet kokius smulkius vaizdus, kurie galėjo būti prarasti žemesniuose kodavimo sluoksniuose [19].

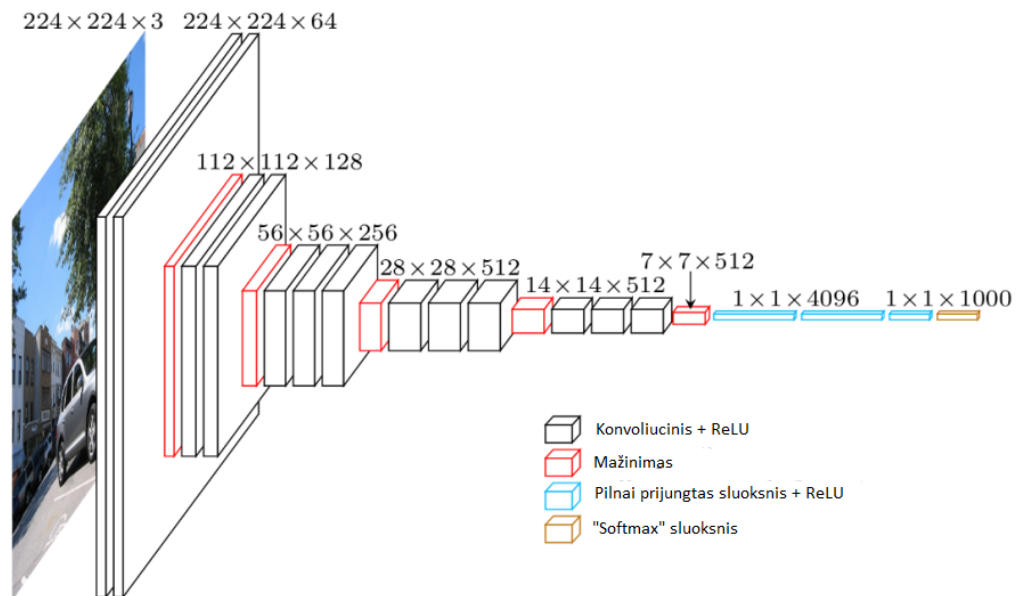
Paduodamuose į tinklą vaizduose gali būti vienas ar daugiau įtrūkių, todėl išėjime turimas segmentacijos žemėlapis turi būti tokio pat dydžio, kuriame atvaizduojamos tikimybės kiekvienai tinkle išskirtai grupei. Kadangi sprendžiant įtrūkimų problemas, turimos grupės yra tik dvi, išėjime turimas segmentacijos žemėlapis kuriame pikseliai su dangos įtrūkiais atvaizduojami juodai, o be jų – baltai arba atvirkščiai.

1.6. Žinių perkėlimas

- **VGG-16**

VGG-16 yra konvoliucinis neuroninis tinklas, sukurtas K. Simonyan ir A. Zisserman iš Oxfordo universiteto. Jis yra apmokytas iš daugiau kaip 14 milijonų paveikslėlių turinčios „ImageNet“ duomenų bazės. Lyginant su kitais, su šia duomenų baze, iš anksto apmokytais tinklais, šis pasiekia 90,1% tikslumą.

VGG-16 tinklas yra sudarytas iš 16 sluoksnių (žr. 5 pav.), ir turi beveik 140 milijonų svorinių parametrų, bei yra pajėgus klasifikuoti vaizdus į 1000 klasių [20].



5 pav. VGG-16 tinklo architektūra

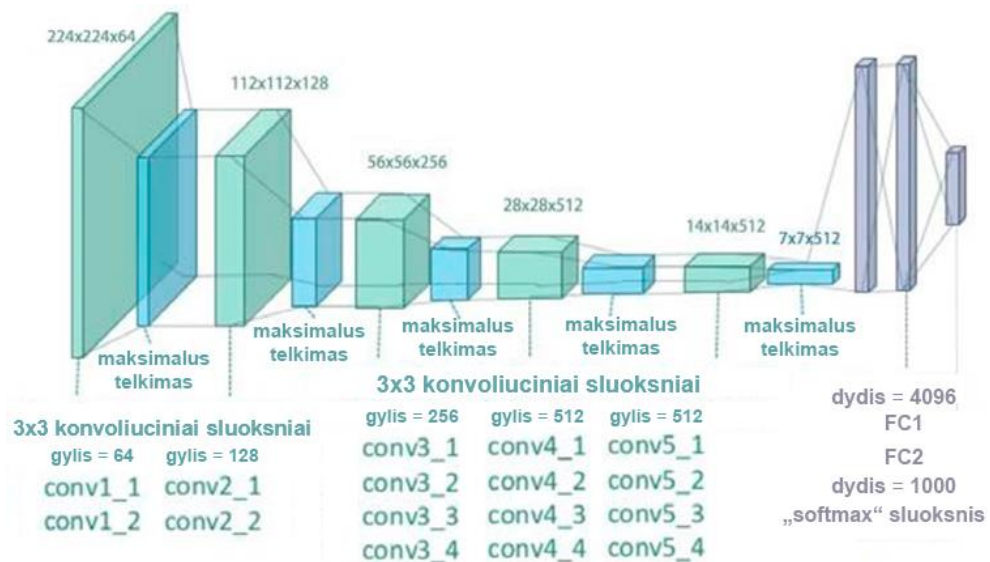
Tinklo įėjimas yra nustatyto 224 x 224 dydžio RGB paveikslėlis, kuris praleidžiamas per konvoliucinius tinklo sluoksnius, kuriuose yra panaudoti labai maži 3 x 3 dydžio filtrai. Maksimalus telkimas atliekamas 2 x 2 pixelio langu ir žingsniu lygiu 2. Pirmieji du pilnai sujungti sluoksniai yra sujungti 4096 kanalais, o trečiasis yra sujungtas 1000 kanalų, kiekvienai klasei po vieną. Visi paslėpti sluoksniai yra aktyvuojami ReLU nelinejine funkcija. Taip pat, panaudota vietinio atsako normalizacija (LRN), kuri nedaro įtakos tinklo įverčiams, bet sumažina naudojamus resursus ir skaičiavimo laiką [21].

- **VGG-19**

VGG-19 kaip ir VGG-16 yra konvoliucinis neuroninis tinkas, sukurtas Oxfordo universitete ir iš anksto apmokytas su „ImageNet“ duomenų baze. VGG-19 tinklo fiksuotas įėjimas yra 224 x 224 dydžio RGB paveikslėlis, šis tinklas sudarytas iš 19 sluoksnių (žr. 6 pav.) – 16 konvoliucinių ir 3 pilnai sujungtų sluoksnių. Kaip ir VGG-16, taip ir VGG-19 konvoliuciniai sluoksniai turi 3 x 3 dydžio filtrą, o maksimalaus telkimo sluoksniai turi 2 x 2 pixelio langą su žingsniu lygiu 2 [22].

Skirtumai tarp VGG-16 ir VGG-19 konvoliucinių neuroninių tinklų [23;24]:

- VGG-19 tinklas sudarytas iš 19 gilaus neuroninio tinklo sluoksnių, kai tuo tarpu VGG-16 sudarytas atitinkamai iš 16 sluoksnių. VGG-19 turi 3 konvoliuciniais sluoksniais daugiau, nei VGG-16;
- VGG-19 turi beveik 5,5 milijonų daugiau svorinių parametrų lyginant su VGG-16;
- VGG-19 turi 90% tikslumą, 1 dešimtąja dalimi mažesnę už VGG-16.



6 pav. VGG-19 tinklo architektūra

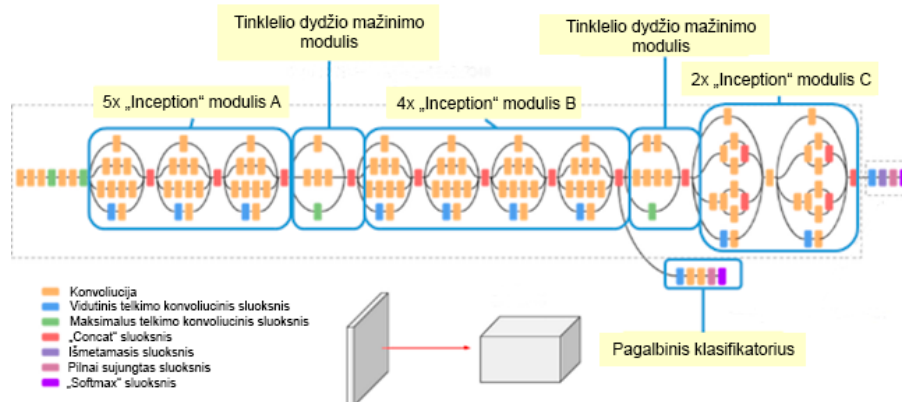
- **InceptionV3**

Vaizdų atpažinimui pasaulyje plačiai naudojamas InceptionV3 gilusis konvoliucinis neuroninis tinklas sudarytas iš 48 sluoksnių. 2015 metais ILSVRC konkurse pristatytas ir iškovojo pirmąją vietą su 93,7% top 5 tikslumu. Kuriant šį tinklą, dėmesys buvo nukreiptas į dideles resursų sąnaudas, norint naudoti konvoliucinius tinklus. Stengiantis sumažinti resursus, jie nusprendė pasinaudoti mintimi analizuoti prieš tai buvusio sluoksnio koreliaciją, aukščiausią koreliaciją turinčius klasterizuoti į grupes. Šios grupės apjungia prieš tai buvusį ir būsimą sluoksnį. Manoma, kad kiekvienas aukštesnis sluoksnis atsakingas už tam tikrą įvesties vaizdo sritį [25].

Fiksuotas įėjimo dydis yra 299 x 299 pikselio RGB paveikslėlis. Šis tinklas, kaip ir kiti, apmokytas su „ImageNet“ duomenų baze, iš kurios išsiaugino beveik 24 milijonus svorinių parametrų ir gali klasifikuoti paveikslėlius į 1000 klasių. Pats modelis pasižymi savo neįprasta struktūra (žr. 1 lentelė) sudaryta iš nuoseklių ir lygiagrečių konstrukcinių blokų (žr. 7 pav.) [26].

1 lentelė. InceptionV3 tinklo struktūra.

Sluoksnis	Tinkelio dydis/žingsnis	Įvesties dydis
Konvoliucinis	3x3/2	224x224x3
Konvoliucinis	3x3/1	111x111x32
Konvoliucinis padidintojas	3x3/1	109x109x32
Telkimas	3x3/2	109x109x64
Konvoliucinis	3x3/1	54x54x64
Konvoliucinis	3x3/2	52x52x80
Konvoliucinis	3x3/1	25x25x192
„Inception“ modulis A x3	-	25x25x288
„Inception“ modulis B x5	-	12x12x768
„Inception“ modulis C x2	-	5x5x1,280
Pilnai sujungtas	51,200x1,024	5x5x2,048
Pilnai sujungtas	1,024x1,024	1,024
Pilnai sujungtas	1,024x4	1,024
„Softmax“ sluoksnis	Klasifikatorius	4



7 pav. InceptionV3 tinklo architektūra

Tinklas suskaidytas į tris puikiai sukurtus modulius Inception A, Inception B ir Inception C, visi moduliai gali ištraukti požymius ir tuo pačiu sumažinti parametų skaičių. Kiekvienas iš jų yra sudarytas iš kelių konvoliucinių ir lygiagrečių telkimo sluoksnių. Konvoliuciniai sluoksniai su mažu filtro ekranu tikiu kaip 3×3 , 1×3 , 3×1 , ar 1×1 yra naudojami parametų skaičiaus mažinimui [27].

- **MobileNet**

Konvoliuciniai neuroniniai tinklai tapo efektyvia ir gerai išvystyta technika, leidžiančia pasiekti aukščiausių rezultatų daugelyje sričių. Tačiau norint gauti didesnę tikslumą, konvoliucinių tinklų struktūra būna didesnė ar gilesnė ir tampa neįmanoma panaudoti jų ribotų išteklių įrenginiuose, tokiuose kaip mobilieji telefonai ar namų robotai. Norint išspręsti šią problemą, google mokslininkų komanda, kurdama MobileNet tinklo modelį stengėsi sukurti tinklą, kuriam būtų reikalinga mažiau resursų. Jie standartines konvoliucinių tinklų operacijas pakeitė į vaizdo gyliui jautrius konvoliucinius sluoksnius, po kurių seka keli paprasti 1×1 konvoliuciniai sluoksniai ir faktorizavimas [28].

MobileNet konvoliucinis neuroninis tinklas sukurtas Andrew G. Howard'o komandos 2017 metais. Šis tinklas gali apdoroti daugybę užduočių tokių kaip objektų atpažinimas, veido bruožų atpažinimas ar didesnio masto geografinė lokalizacija. Be to, jam buvo sukurtas globalinis kintamasis, leidžiantis pasirinkti kokią dalį tinklo naudoti, su juo galima laviruoti tarp greičio ir tikslumo verčių [29].

Tinklo architektūra pateikta 2 lentelėje. Tinklas turi 28 sluoksnius, po kiekvieno sluoksnio seka paketo normalizavimas ir nelinijinė ReLU aktyvavimo funkcija. Pirmasis tinklo sluoksnis yra pilnai konvoliucinis ir yra fiksuoto 224×224 RGB įėjimo. Šis tinklas yra apmokytas „ImageNet“ duomenų baze ir turi kiek daugiau nei 4 milijonus parametų, tad yra gerokai mažesnis už kitus pristatytus tinklus, tačiau pasiekia 0,895% top 5 tikslumą. Lyginant su Vgg19, jis yra beveik 34 kartus mažesnis, kad reiškia ir tai, jog jis yra mobilesnis [29].

2 lentelė. MobileNet tinklo struktūra

Tinkelio tipas / žingsnis	Filtro forma	Įvesties dydis
Konvoliucinis / s2	$3 \times 3 \times 3 \times 32$	$224 \times 224 \times 3$
Konvoliucinis dw / s1	$3 \times 3 \times 32$ dw	$112 \times 112 \times 32$
Konvoliucinis / s1	$1 \times 1 \times 32 \times 64$	$112 \times 112 \times 32$
Konvoliucinis dw / s2	$3 \times 3 \times 64$ dw	$112 \times 112 \times 64$
Konvoliucinis / s1	$1 \times 1 \times 64 \times 128$	$56 \times 56 \times 64$
Konvoliucinis dw / s1	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Konvoliucinis / s1	$1 \times 1 \times 128 \times 128$	$56 \times 56 \times 128$
Konvoliucinis dw / s2	$3 \times 3 \times 128$ dw	$56 \times 56 \times 128$
Konvoliucinis / s1	$1 \times 1 \times 128 \times 256$	$28 \times 28 \times 128$
Konvoliucinis dw / s1	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Konvoliucinis / s1	$1 \times 1 \times 256 \times 256$	$28 \times 28 \times 256$
Konvoliucinis dw / s2	$3 \times 3 \times 256$ dw	$28 \times 28 \times 256$
Konvoliucinis / s1	$1 \times 1 \times 256 \times 512$	$14 \times 14 \times 256$
5× Konvoliucinis dw / s1	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Konvoliucinis / s1	$1 \times 1 \times 512 \times 512$	$14 \times 14 \times 512$
Konvoliucinis dw / s2	$3 \times 3 \times 512$ dw	$14 \times 14 \times 512$
Konvoliucinis / s1	$1 \times 1 \times 512 \times 1024$	$7 \times 7 \times 512$
Konvoliucinis dw / s2	$3 \times 3 \times 1024$ dw	$7 \times 7 \times 1024$
Konvoliucinis / s1	$1 \times 1 \times 1024 \times 1024$	$7 \times 7 \times 1024$
Vidutinis telkimas / s1	Telkimas 7×7	$7 \times 7 \times 1024$
Pilnai sujungtas / s1	1024×1000	$1 \times 1 \times 1024$
„Softmax“ / s1	Klasifikatorius	$1 \times 1 \times 1000$

Konvoliucinis dw (angl. „Depthwise“) – konvoliucinis gyliui jautrus sluoksnius

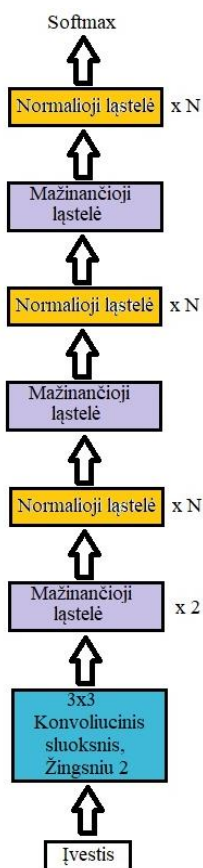
- **NASNet**

Kuriant neuroninio tinklo architektūrą, norint pasiekti geriausius rezultatus, reikia įdėti daug pastangų ir žinių į šį procesą, tam yra buriamos komandos mokslininkų, ieškančių naujų sprendimų. Todėl šis tinklas, skirtingai nuo kitų, to nereikalavo. Tam nereikėjo burti komandos, kadangi ši struktūra sukurta ne žmogaus. NASNet architektūra rasta automatiškai AutoML sistemos. Šis tinklas pats ieško geriausio būdo savo struktūros sudarymui, tam naudodamas didžiulius Google skaičiavimo resursus [30].

Pagrindiniai NASNet komponentai yra dviejų tipų ląstelės: normalioji ir mažinančioji. Mažinančioji ląstelė sumažina per pusę požymių žemėlapiu aukštį ir plotį, kuris praleidžiamas pro ją. Priešingai jai, normalioji ląstelė išlaiko šias dvi dimensijas tokias pačias, kokios yra prieš ją [31].

Bendroji NASNet struktūra (žr. 8 pav.) yra sukurta sudedant N normalių ląstelių į tarpą tarp dviejų mažinančiųjų. Yra du tipai NASNet tinklų, NASNet-large kuriame N normalių ląstelių yra šešios ir NASNet-mobile kur N yra lygus 4, skirtas dirbti su ribotų resursų įrenginiais. NASNet-large tinklas pasiekia 96% top 5 tikslumą „ImageNet“ duomenų bazėje ir turi beveik 89 milijonus parametrų, o NASNet-mobile pasiekia 91,9% top 5 tikslumą ir turi tik 5,3 milijono parametrų [31].

Tiek normalioji, tiek mažinančioji ląstelė yra sudaryta kitų blokų, kurie susideda iš populiariųjų operacijų naudojamų konvoliuciniuose neuroniniuose tinkluose, tokių kaip: konvoliucijos, maksimalus telkimas, vidutinis telkimas, išplėstų konvoliucijų ar vaizdo gyliui jautrių konvoliucijų [32].

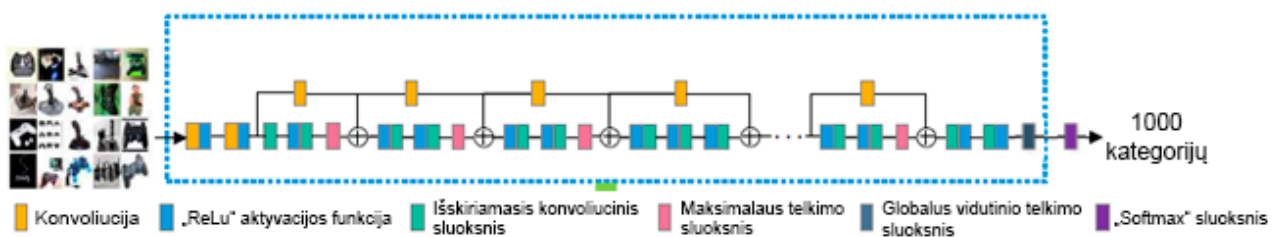


8 pav. NASNet struktūra

- **Xception**

Xception – tai 2016 metais Francois Chollet sukurtas konvoliucinis neuroninis tinklas. Šis tinklas grindžiamas prielaida, kad koreliacija tarp įvesties kanalų yra visiškai nepriklausomi nuo erdvinės koreliacijos. Xception, tai Inception architektūros pagrindu sukurtas tinklas, kuriame panaudota naujesnė technika, kuri savo veikimu panaši į „MobileNet“ vaizdo gyliui jautrių sluoksnių logikai, kai erdvinės konvoliucijos sukuriamos atskirai kiekvienam kanalui, o po jų seka 1 x 1 standartinis konvoliucinis sluoksnis, kad būtų užtikrintas pilnas padengimas. Taip ir Xception atskirtos erdvinės ir giluminės koreliacijos ir sprendžiamos atskirai, kadangi standartiniai konvoliuciniai sluoksniai buvo pakeisti vaizdo gyliui jautriais konvoliuciniais sluoksniais. Šis atskyrimas leido sumažinti skaičiavimo resursus ir darbinės atminties poreikį [33].

Tinklas sudarytas iš 36 konvoliucinių sluoksnių suskirstytų į 14 modulių, kurie sujungti nuosekliu likutiniu būdu, išskyrus pirmąjį ir paskutinį modulius (žr. 9 pav.). Šis jungimas padeda spręsti nykstančio gradiento problemą sukuriant nuorodas tinkle. Nuorodos leidžia prieš tai buvusio sluoksnio išvestį, panaudojus sumavimo operaciją, naudoti kito sluoksnio įvesčiai. Xception yra tvirtesnis ir galingesnis lyginant su Inception tinklu, nors dokumentacija abiejų yra panaši. Xception turi beveik 23 milijonus parametrų, o top 5 tikslumas yra 94,5% „ImageNet“ duomenų bazėje [34].



9 pav. Xception tinklo architektūra

1.7. Modelio vertinimo įverčiai

Konvoliucinio neuroninio tinklo modelio tikslumui ir korektiškumui nustatyti, kaip ir visur, yra naudojami tam tikri įverčiai. Visi pagrindiniai tam skirti vertinimo įverčiai yra sudaryti remiantis sumaišymų matrica (angl. *Confusion Matrix*). Tai speciali atsitiktinųjų lentelės rūšis susidedanti iš dviejų eilučių ir dviejų stulpelių, kurie aprašo tikslą ir spėjimą. Matricos pavyzdys matomas pateiktoje 3 lentelėje [35].

3 lentelė. Sumaišymų matrica

	Teigiama A	Neigiama $\neg A$
Teisinga B	$A \cap B$	$\neg A \cap B$
Neteisinga $\neg B$	$A \cap \neg B$	$\neg A \cap \neg B$

1.7.1. Tikslumas

Tikslumas (angl. *Precision*), kuris aprašomas 1 formule. Šis įvertis skirtas vertinti santykį tarp teigiamų atsakymų ir visų rezultatų. Jis vertina tik teigiamą pusę, todėl skirtas paieškos sistemoms, tačiau klaidingas klasifikavimo sistemoms. Šis įvertis parodo, kokia dalis klasei priskirtų įrašų jai iš tikrųjų priklauso [36].

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

Čia TP žymi klasei teisingai priskirtų jos įrašų poaibį (angl. *True positive*);

FP žymi klasei neteisingai priskirtų kitos klasės įrašų poaibį (angl. *False positive*);

TN žymi teisingai nepriskirtų kitos klasės įrašų poaibį (angl. *True negative*);

FN žymi klasei neteisingai nepriskirtų įrašų poaibį (angl. *False negative*);

1.7.2. Atkūrimas

Atkūrimas (angl. *Recall*) aprašomas 2 formule. Šis įvertis vertina santykį tarp teisingai suklasifikuotų ir visų gautų teigiamų reikšmių. Parodo kokia dalis klasės įrašų buvo teisingai atpažinta [36].

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

1.7.3. Taiklumas

Taiklumas (angl. *Accuracy*) aprašomas 3 formule. Šis įvertis vertina santykį tarp visų teisingai suklasifikuotų klasių ir visų duomenų. Jis leidžia matyti viso modelio bendrą tikslumą, tačiau turėtų būti vengiamas, jeigu teigiamų ir neigiamų klasių imtys labai skiriasi.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (3)$$

1.7.4. F_1

F_1 įvertis yra tikslumo (1) ir atkūrimo (2) harmoninis vidurkis, skirtas apibendrinti abu įverčius. Jis aprašomas (4) formule, skirta, kai abu matai turi vienodą svorį. Kitu atveju naudojama (5) formulė, kai yra reguliuojama šių įverčių svarba. F_1 galima apskaičiuoti kiekvienai klasei atskirai, tai aktualu, kai viena iš vertinamų klasių laikoma svarbesne už kitą [36].

$$F_1 = \frac{2TP}{2TP + FP + FN} = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (4)$$

$$F_\beta = (1 + \beta^2) * \frac{Precision * Recall}{(\beta^2 * Precision) + Recall} \quad (5)$$

1.7.5. IoU

IoU įvertis, dar vadinamas Džakardo indeksu (angl. *Jaccard index*). Jis nurodo, kokia dalis detektoriaus regiono aibės sutampa su tikrosiomis objekto ribomis. Taip įvertinant objektų aptikimo tikslumą. Dažniausiai, esant $IoU > 0.5$ spėjimas laikomas teisingu. Tai aprašoma (6) formule [37].

$$IoU = \frac{TP}{FP + TP + FN} \quad (6)$$

1.7.6. MCC

Dirbtinių neuroninių tinklų, sudarytų iš dviejų klasių, kokybės apibūdinimui naudojamas Matthews koreliacijos koeficientas, 1975 metais pristatytas biochemiko Brian. W. Matthews. Tai tikslo ir prognozės koeficientas, kuris gali būti naudojamas ir tada, kai klasių disbalansas yra labai didelis. Koeficientas aprašomas (7) formule, jo reikšmė svyruoja $\{-1..1\}$, kur 1 yra idealus klasifikavimas, 0 – spėlionės, o -1 visiškai neteisingas.[41].

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (7)$$

2. Duomenų bazė

Esant konkrečiai problemai, yra galimybė duomenų bazę susidaryti patiems. Tačiau, norint tai padaryti kokybiškai, reikia investuoti daug laiko ir nemažai resursų. Todėl, šiame darbe buvo nuspręsta naudoti vieną iš jau sudarytų duomenų bazių – „SDNET2018“, kuri yra tinkama iškeltai problemai spręsti [38].

Duomenų bazė SDNET2018 yra sudaryta iš 56000 anotuotų tilto šaligatvių, kelio dangos ir tilto pagrindo įtrūkusio ir sveiko betono paveikslėlių (žr. 10 pav.). Sukurta būtent tam, kad būtų galima apmokyti, patikrinti, testuoti, bei palyginti tarpusavyje konvoliucinius neuroninius tinklus, ar kitus vaizdo apdorojimo algoritmus.



10 pav. SDNET2018 duomenų bazė sudaryta iš (a)-smulkių įtrūkimų, (b)-gilių trūkių, (c)-šėšėlių, (d)-dėmių, (e)-įtrūkimų grubiame paviršiuje, (f)-tuštumos, (g)-briaunų, (h)- jungčių ir iškorėjusių paviršių, (i)-atsitiktinių fono kliūčių;

Tam buvo panaudotas Nikon 16 megapixelių fotoaparatas, su kuriuo 50 cm atstumu buvo fotografuojamas paviršius. Paviršiaus apšviestumas svyravo nuo 1500 iki 3000 liumėnų. Vienos padarytos nuotraukos dydis buvo 4068 x 3456 pikselių, kurių kiekviena apytikriai atspindi 100 x 85 cm paviršiaus plotą. Tada, kiekviena iš jų susegmentuota į 256 x 256 pikselio paveikslėlius, atvaizduojančius apytiksliai 6 x 6 cm paviršiaus plotą. Iš viso to gavosi 56 tūkst. skirtingų paveikslėlių, kurie buvo sugrupuoti į 3 grupes - šaligatvių, kelio dangos ir tilto pagrindo, bei kiekviena iš jų į 2 pogrupius – su įtrūkimais ir be jų. Aprašymas 3 lentelėje.

Šią duomenų bazę sudarė Jutos valstijos universiteto prašymu, tačiau ją leidžiama naudoti laisvai akademiniais tikslais.

4 lentelė. SDNET2018 statistinis aprašas.

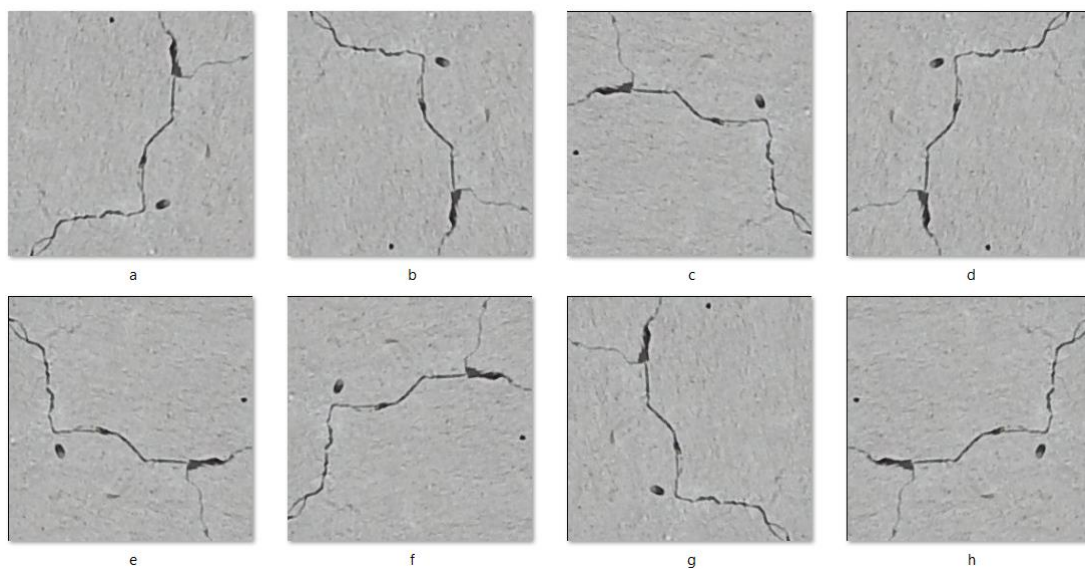
	Su įtrūkimais	Be įtrūkimų	Viso:
Kelio danga (D)	2025	11595	13620
Tilto pagrindas (W)	3851	14287	18138
Šaligatvis (P)	2608	21726	24334
Viso:	8484	47608	56092

2.1. Augmentacija

Gilieji konvoliuciniai neuroniniai tinklai buvo sėkmingai pritaikyti vaizdų analizės užduotims, tokioms kaip vaizdų atpažinimas ir klasifikavimas, ar vaizdų segmentavimas. Tačiau, šie tinklai yra labai priklausomi nuo duomenų imties ir jos dydžio, kitaip tariant, kuo didesnė ir kokybiškesnė duomenų bazė – tuo geresni rezultatai gaunami. Deja, daugelyje taikymo sričių, tokių kaip medicina, nėra prieigos prie didelių duomenų bazių, todėl reikia ieškoti sprendimo, norint ir toliau vystyti algoritmus šioje srityje [39].

Tam, kad kurti naudingus giliojo mokymosi modelius, validavimo paklaida turi mažėti kartu su mokymo paklaida, kitu atveju tinklas laikomas permokytu (angl. *Overfitting*). Šioje vietoje, į pagalbą pasitelkiama duomenų bazės augmentacijos galimybė. Duomenų bazės augmentacija apima daug technikų duomenų praplėtimui, tam naudojama tokios kaip geometrinio ar spalvų transformavimo, atsitiktinio ištrynimo, kernelio filtrų ar vaizdų sumaišymo algoritmai [40].

Šiame darbe naudojamos duomenų bazės problema yra jos netolygumas, kadangi ji sudaryta fotografuojant visą tiltą, todėl beveik 6 kartus didesnę dalį sudaro nuotraukos be įtrūkimų. Norint, kad tinklas mokytųsi, reikėjo duomenis suvienodinti, todėl buvo panaudotas tradicinis duomenų augmentacijos metodas, kurio metu duomenų bazės „įtrūkimų“ imtis buvo išplėsta 8 kartus, tai yra, iš vieno paveikslėlio, buvo gauti 8 skirtingi paveikslėliai (žr. 11 pav).



11 pav. a – originalus vaizdas, b – apverstas, c – pasuktas 90° , d – pasuktas 180° , e – pasuktas 270° , f – apverstas originalus ir pasuktas 90° , g - apverstas originalus ir pasuktas 180° , h – apverstas originalus ir pasuktas 270°

3. Eksperimentinė dalis

Automatizuotam vizualiniam defektų aptikimui nuspręsta naudoti giliuosius konvoliucinius tinklus iš „Keras“ duomenų bazės [12]. Pasirinkti 6 gilieji neuroniniai tinklai: VGG16, VGG19, InceptionV3, MobileNet, NASNet-mobile ir Xception.

Šie tinklai, su savo unikaliais svoriniais koeficientais, buvo panaudoti kaip požymių ištraukėjai, siekiant iširti jų gebėjimą atpažinti defektus, šiuo atveju, tai buvo vaizdai iš SDNET2018 duomenų bazės su 3 tekstūrų paviršiais. Visos trys tekstūros yra atskirtos ir papildomai atskirtos jų sveikos ir įtrūkusios vietos. Taip pat, visos grupės ir pogrupiai buvo suskaidyti į mokymo, validavimo ir testavimo imtis atitinkamai 60% , 10% ir 30%.

Nuspręsta šiuos tinklus išnagrinėti keliais būdais:

- Pirmasis būdas – patikrinti kaip jie geba atskirti skirtingų tekstūrų paviršius;
- Antrasis – iširti, kaip tiksliai šie tinklai gali atskirti visokių formų ir dydžių įtrūkimus skirtingose struktūrose;
- Trečiasis – pritaikyti U-Net tipo tinklą, palyginti gautus segmentavimo žemėlapius su atitinkančiomis nuotraukomis iš naudojamos duomenų bazės.

Kadangi šie tinklai yra pritaikyti 1000 skirtingų klasių, o šiems tyrimams reikalingos pirmu atveju 3 klasės, o antru atveju 2 klasės, dėl to reikėjo modifikuoti visus tinklus. Modifikacijos atliktos visiems vienodos norint palyginti rezultatus. Taip pat, visiems tinklams buvo pridėti keli sluoksniai pabaigoje, taip perdarant klasifikatorių į reikiamą šiam tyrimui.

3.1. Tekstūros atpažinimas

Kadangi duomenų bazė yra sudaryta iš trijų skirtingų paviršių nuotraukų, tai šiuo būdu buvo tiriama, kaip tinklai sugeba atskirti vieną paviršių nuo kito. Gauti rezultatai pateikti 5 lentelėje.

5 lentelė. Tekstūros atpažinimo rezultatai.

	F ₁	Atkūrimas (angl. <i>Recall</i>)	Iteracijų sk.	Sumaišymo matrica
VGG16 224x224	0,24	0,33	30	[4415 0 0] [8066 0 0] [6061 0 0]
VGG19 224x224	0,33	0,33	30	[0 0 4415] [0 0 8066] [0 0 6061]
InceptionV3 299x299	0,95	0,93	30	[3826 294 295] [89 7969 8] [157 160 5744]
MobileNet 224x224	0,68	0,71	30	[3245 0 1170] [3793 3615 658] [268 0 5793]
NASNetMobile 224x224	0,91	0,9	30	[3433 142 840] [113 7510 443] [81 18 5962]
Xception 299x299	0,94	0,94	30	[4013 241 161] [84 7982 0] [320 211 5530]

3.1.1. VGG16

Bandant apmokyti VGG16 konvoliucinį tinklą, jis nesugebėjo atskirti paviršių ir visus įvesties paveikslėlius priskyrė vienai klasei. Buvo padaryta prielaida, kad tinklas nesugeba priimti mokymo, dėl netolygaus duomenų pasiskirstymo, dėl to buvo taikoma duomenų augmentacija mokymo imčiai ir taip klasės sulygintos iki beveik lygių. Taip pat, į programos kodą buvo įtrauktos funkcijos duomenų suvienodinimui, tam kad klasės pagal savo apimtį turėtų svorinį koeficientą ir tinklas į tai reaguotų.

Tačiau, po pakeitimų VGG16 rezultatai nepasikeitė. Todėl, po 30 iteracijų mokymo, jo F1 įvertis yra 0,24, su 0,06 preciziškumu, kas reiškia, kad sumaišymo matricoje visi duomenys yra priskirti tik vienai klasei – „kelio danga“.

3.1.2. VGG19

VGG19 tinklui apmokyti, jau buvo paruoštas pagrindas ir jis mokytas su augmentuota duomenų baze ir modifikuotu pirminiu kodu. Mokymo eigoje, stebint mokymosi klaidos kitimą, atrodė, kad tinklas mokosi. Tačiau, išlaukus pabaigos tapo aišku, kad kitokia šio tinklo struktūra, lyginant su VGG16, naudos tinklo rezultatams nedavė.

Po 30 iteracijų mokymo, tinklo F1 įvertis yra 0,33, su 0,11 preciziškumu. Tinklo sumaišymo matricoje buvo aiškiai matoma, kad tinklas nesimoko, o tik priskiria testinius duomenis vienai klasei,

šiuo atveju priskiriama – „tilto pagrindas“ klasei. F1 įverčio skirtumas su VGG16 F1 reikšme atsirado, kadangi duomenys testavimo imtyje negalėjo būti augmentuojami, todėl vaizdų kiekis tarp klasių šiek tiek skiriasi.

3.1.3. MobileNet

MobileNet tinklo mokymo metu, jau mokymo eigoje buvo matoma, kad tinklas duos geresnius rezultatus, nei VGG16 ar VGG19. Po 30 iteracijų mokymo, tinklo F1 koeficientas išaugo iki 0,68, su 0,79 preciziškumu. MobileNet sumaišymo matricoje aiškiai matomi geresni rezultatai, tinklas sugeba atpažinti dalį vaizdų su 68% bendru tinklo tikslumu. Kelio dangą tinklas atpažįsta 55%, šaligatvius 62%, o sienas 85%.

3.1.4. NASNet-mobile

NasNet-mobile nors ir skirtas ribotų resursų įrenginiams, bet jo mokymo laikas ilgesnis, lyginant su kitais tirtais tinklais. Po ilgo mokymo, susidedančio iš 30 iteracijų, tinklas įgijo 0,91 F1 įvertį su 0,92 preciziškumu. Taip pat, tinklo sumaišymo matricoje matoma, kad NASNet-mobile geba tiksliau klasifikuoti paviršiaus tekstūras. Šis tinklas pasiekė bendrą 91% tikslumą. Kelio dangą jis atpažįsta 85%, šaligatvius 95%, o sienas 90%.

3.1.5. InceptionV3

Šis InceptionV3 konvoliucinis tinklas, po 30 iteracijų mokymo, parodė išpūdingus rezultatus. Šio tinklo tikslumas siekė 95%, jis gebėjo atpažinti kelio dangą su 90%, šaligatvius su 97%, o sienas su 95% tikslumais. InceptionV3 įgijo 0,95 F1 įvertį ir 0,95 preciziškumą.

3.1.6. Xception

Xception nedaug atsiliko nuo InceptionV3 ir po 30 iteracijų mokymo, įgijo 0,94 F1 įvertį su 0,95 preciziškumu. Analizavus tinklo sumaišymo matricą, matoma, kad jis puikiai tvarkėsi su užduotimi ir tik InceptionV3 tinklo truputį didesnis sienų atpažinimo pranašumas ir klasės svoriai nusvėrė rezultata į savo pusę.

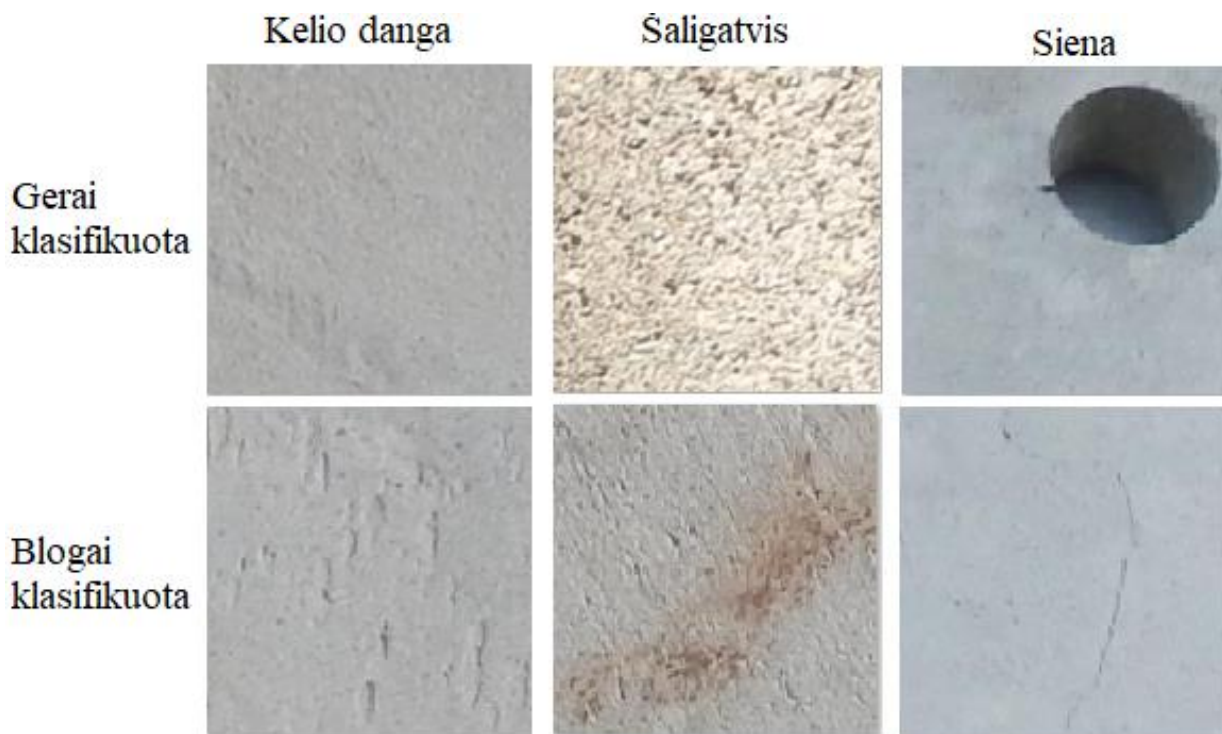
3.1.7. Išvada

- Ištyrus šiuos tinklus, buvo nustatyta, kad geriausiai paviršiaus tekstūrą klasifikuoja InceptionV3 konvoliucinis tinklas.

6 lentelė. InceptionV3 tinklo paviršiaus tekstūros sumaišymo matrica

	Kelio danga	Šaligatviai	Sienos
Kelio danga	3826	294	295
Šaligatviai	89	7969	8
Sienos	157	160	5744

- Klasifikuodamas InceptionV3 geriausiai atskiria šaligatvius, o labiausiai maišo kelio dangos tekstūrą su šaligatviais bei sienomis.



12 pav. InceptionV3 suklasifikuotų nuotraukų pavyzdys

- Tikėtina, kad tinklas didelį dėmesį skyrė šaligatvio požymiams, kadangi kaip pavaizduota nuotraukoje, dangos panašios, nėra kažkokių labai aiškių skirtumų, o viskas priskiriama šaligatviams.

3.2. Defekto atpažinimas

Antrojo tyrimo metu, tinklai buvo mokytį aptikti įtrūkimą skirtingos struktūros paviršiuose, kai tekstūra yra žinoma. Rezultatai pateikti 7, 8 ir 9 lentelėse ir trumpai apžvelgti.

7 lentelė. Sienos defekto atpažinimo rezultatai.

	F ₁	MCC	Iteracijų sk.	Sumaišymo matrica
VGG16 224x224	0,21	0	30	[1138 0] [4262 0]
VGG19 224x224	0,21	0	30	[1138 0] [4262 0]
InceptionV3 299x299	0,77	0,07	30	[114 1024] [236 4026]
MobileNet 224x224	0,8	0,43	30	[678 460] [606 3656]
NASNetMobile 224x224	0,6	0,32	30	[984 154] [2013 2249]
Xception 299x299	0,87	0,56	30	[489 649] [66 4196]

8 lentelė. Kelio dangos defekto atpažinimo rezultatai.

	F ₁	MCC	Iteracijų sk.	Sumaišymo matrica
VGG16 224x224	0,15	0	30	[603 0] [3452 0]
VGG19 224x224	0,15	0	30	[603 0] [3452 0]
InceptionV3 299x299	0,87	0,37	30	[175 428] [99 3353]
MobileNet 224x224	0,86	0,25	30	[57 546] [10 3442]
NASNetMobile 224x224	0,15	0	30	[603 0] [3452 0]
Xception 299x299	0,88	0,42	30	[169 434] [48 3404]

9 lentelė. Šaligatvio defekto atpažinimo rezultatai.

	F ₁	MCC	Iteracijų sk.	Sumaišymo matrica
VGG16 224x224	0,11	0	30	[784 0] [6514 0]
VGG19 224x224	0,11	0	30	[784 0] [6514 0]
InceptionV3 299x299	0,74	0,26	30	[490 294] [1580 4934]
MobileNet 224x224	0,93	0,59	30	[348 436] [51 6463]
NASNetMobile 224x224	0,78	0,28	30	[458 326] [1262 5252]
Xception 299x299	0,89	0	30	[0 784] [0 6514]

3.2.1. VGG16

Bandant apmokyti VGG16 neuroninį tinklą siekiant atpažinti struktūros įtrūkimus, tinklas mokymo nepriėmė. Po 30 mokymo iteracijų, siekiant atpažinti plyšius sienų tekstūros paviršiuje, įgytas 0,21 F1 įvertis, o MCC yra lygus 0, kas reiškia, kad tinklas yra neįgalus atpažinti defektui šiame paviršiuje.

Siekiant apmokyti tinklą darbui su kelio dangos įtrūkimais, tinklas įgijo 0,15 F1 įvertį ir MCC lygu 0, kas taip pat reiškia, kad tinklas nesugeba dirbti ir su šia danga.

Pabandžius tinklą apmokyti atpažinti defektą šaligatvio paviršiuje, po 30 mokymo iteracijų tinklas įgijo 0,11 F1 įvertį ir MCC lygų 0, kas galutinai parodė, jog tinklas yra absoliučiai neįgalus šiam darbui.

3.2.2. VGG19

Apmokius VGG19 tinklą darbui su sienose atsiradusiais įtrūkimais, tinklas po 30 mokymo iteracijų pasiekė 0,21 F1 įvertį ir MCC lygų 0. Jis kaip ir VGG16 neišmoko atpažinti defektų šioje tekstūroje.

Bandant tinklą apmokyti darbui su kitomis tekstūromis, tinklas taip pat nesimokė ir visus vaizdus priskyrė vienai klasei. Todėl VGG19, kaip ir VGG16 yra visiškai neįgalus dirbti nei su viena iš šių dangų.

3.2.3. MobileNet

Pradėjus mokyti MobileNet tinklą darbui su šaligatviais iš karto matėsi, kad rezultatai bus kitokie, nei pastarųjų dviejų tinklų. Po 30 mokymo iteracijų tinklas įgijo stebinantį 0,93 F1 įvertį, lydimą 0,59 MCC bei 0,93 preciziškumo. Tinklas sugeba klasifikuoti šaligatvio dangos defektus su 93% bendru tikslumu. Nors, jeigu išskirsčius klasėmis, tai nuotraukas be įtrūkimų šis tinklas klasifikuoja su 96% tikslumu, o su įtrūkimais 56% tikslumu.

Tęsiant tyrimą, tinklas buvo apmokytas darbui su sienų paviršiaus tekstūros įtrūkimais. Po 30 mokymo iteracijų tinklas įgijo 0,80 F1 įvertį, lydimą 0,81 preciziškumo, bei 0,43 MCC koeficiento. Vadinasi, tinklas sugeba klasifikuoti dangos nuotraukas, pagal tai, yra ar nėra įtrūkimo dangos paviršiuje su 80% tikslumu, bet ir čia yra ta pati situacija atkreipus dėmesį į klasių tikslumą. Tinklas nuotraukas be sienų paviršiaus defektų skirsto su 87% tikslumu, o nuotraukas su defektais skirsto su 56% tikslumu.

Blogesnis scenarijus įvyko, bandant tinklą apmokyti darbui su kelio dangos įtrūkimų klasifikavimu. Po 30 mokymo iteracijų, tinklas įgijo 0,86 F1 įvertį, tačiau su 0,25 MCC koeficientu, reiškiančiu, kad pasiskirstymas tarp klasių yra labai didelis. Taip ir yra, tinklas nuotraukas be kelio dangos įtrūkimų klasifikuoja su 93% tikslumu, o nuotraukas su įtrūkimais su 17% tikslumu.

3.2.4. NASNet-mobile

Apmokius NASNet-mobile tinklą dirbti su sienų įtrūkimų klasifikavimu, po 30 mokymo iteracijų tinklas įgijo 0,6 F1 įvertį su 0,81 preciziškumu, o MCC koeficientas yra 0,32. Atkreipiant dėmesį į pasiskirstymą tarp klasių, tai rezultatai nėra labai aukšti, bet jie panašūs abiejuose klasėse, kas taip pat yra naujas pasiekimas tiriant šiuos tinklus.

Tęsiant tyrimą NASNet-mobile apmokytas šaligatvių dangos įtrūkimų klasifikavimui. Po 30 mokymo iteracijų tinklas pasiekė 0,78 F1 įvertį su 0,87 preciziškumu, tačiau MCC yra 0,28, kadangi

yra nemažas pasiskirstymo matricos išsibarstymas. Šį kartą tinklas geriau skirstė įtrūkimus, negu nuotraukas kur jų nėra.

Stengiantis apmokyti tinklą darbui su kelio dangos įtrūkimų klasifikavimu, NASNet-mobile po 30 iteracijų įgijo 0,15 F1 įvertį su 0,02 preciziškumu ir MCC lygiu 0. Todėl, nes šis tinklas absoliučiai nesugeba klasifikuoti kur yra įtrūkimas ir kur jo nėra šioje dangoje.

3.2.5. InceptionV3

Apmokant InceptionV3 tinklą darbui su sienų įtrūkimų klasifikavimu, po 30 mokymo iteracijų šis tinklas įgijo 0,77 F1 įvertį su 0,7 preciziškumu, bet MCC koeficientas siekė vos 0,07. Taip yra dėl to, kad sumaišymo matricoje matoma klasių netolygumas, tinklas puikiai operuoja su nuotraukomis kur nėra įtrūkimų, tačiau blogai skirsto nuotraukas su įtrūkiais, būtent tai ir rodo mažas MCC koeficientas.

Apmokius InceptionV3 dirbti su kelio dangos įtrūkimų klasifikavimu, tinklas po 30 iteracijų įgijo 0,87 F1 įvertį, lydimą 0,85 preciziškumo ir 0,37 MCC koeficiento. Šis tinklas nuotraukas su įtrūkiais klasifikuoja su 40% tikslumu, o sveiko paviršiaus nuotraukas su 93% tikslumu.

Po 30 iteracijų mokymo, siekiant, kad tinklas išmoktų atpažinti šaligatvių tekstūros paviršiaus įtrūkimus, buvo pasiekta 0,74 F1 įvertis su 0,87 preciziškumu, o MCC koeficientas yra 0,26. Žvelgiant į klases, tinklas, nuotraukas su įtrūkiais klasifikavo su 34% tikslumu, o nuotraukas be įtrūkimų su 84% tikslumu. Šis tinklas nemažą dalį testavimo imties nuotraukų be įtrūkimų, priskyrė įtrūkims.

3.2.6. Xception

Siekiant apmokyti Xception neuroninį tinklą, darbui su sienų paviršiaus tekstūros įtrūkiais, po 30 mokymo iteracijų buvo pasiektas 0,87 F1 įvertis su 0,87 preciziškumu, bei 0,56 MCC koeficientu. Šis tinklas lyginant su kitais randa optimaliausią būdą matricos paskirstymui ir balansui tarp klasių. Xception nuotraukas su įtrūkiais skirsto su 58% tikslumu, o be įtrūkimų su 92% tikslumu.

Apmokius Xception tinklą dirbti su kelio dangos įtrūkimų klasifikavimu, tinklas po 30 mokymo iteracijų pasiekė 0,88 F1 įvertį su 0,87 preciziškumu ir 0,42 MCC koeficientu. Tarp klasių nuotraukas su kelio įtrūkiais klasifikavo 41% tikslumu, o be jų su 93% tikslumu.

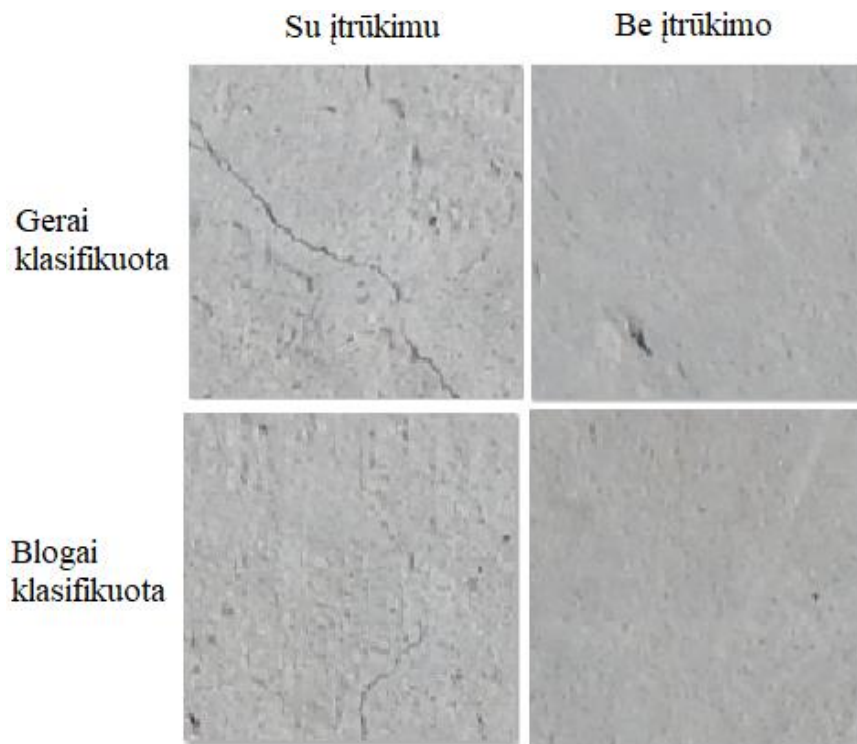
Apmokant tinklą darbui su šaligatvių paviršiaus įtrūkimų klasifikavimu, jau mokymo pradžioje po kelių iteracijų, buvo matoma, kad tinklas mokymo nepriima. Po 30 mokymo iteracijų, tinklas įgijo 0,89 F1, kuris nieko nereiškia, nes MCC buvo lygus 0.

3.2.7. Išvada

- Ištyrus tinklus ir apžvelgus rezultatus, nuspręsta, kad geriausiai, su 0,88 F1 ir 0,42 MCC įverčiais, kelio dangos įtrūkimus klasifikuoja Xception tinklas.

10 lentelė. Xception tinklo kelio dangos įtrūkimų klasifikavimo sumaišymo matrica

	Su įtrūkimu	Be įtrūkimo
Su įtrūkimu	169	434
Be įtrūkimo	48	3404



13 pav. Xception tinklo suklasifikuotų kelio dangos nuotraukų pavyzdys

- Tikėtina, kad tinklas nėra toks pajėgus, atpažinti tokį nežymų įtrūkimą, kaip pavaizduota 13 pav. Taip pat, nuotrauka, kurioje nėra jokio defekto, palaikyta kelio įtrūkimu – tai tinklo klaida.
- Ištyrus pasirinktus dirbtinius neuronius tinklus nustatyta, kad sienų įtrūkimus taip pat geriausiai klasifikuoja Xception neuroninis tinklas, pasiekiantis 0,87 F1 įvertį su 0,56 MCC koeficientu.

11 lentelė. Xception tinklo sienų įtrūkimų klasifikavimo sumaišymo matrica

	Su įtrūkimu	Be įtrūkimo
Su įtrūkimu	489	649
Be įtrūkimo	66	4196



14 pav. Xception tinklo suklasifikuotų sienų nuotraukų pavyzdys

- Tikėtina, kad tinklas nėra toks jautrus, kad atpažinti nežymų įtrūkimą, matomą 14 pav. Taip pat, panašu, kad tinklas maišo žymesnius sienos tekstūros patamsėjimus su įtrūkimais.
- Iš gautų rezultatų nustatyta, kad geriausiai šaligatvių paviršiaus įtrūkimus klasifikuoja MobileNet tinklas, kuris pasiekė 0,93 F1 įvertį su 0,59 MCC koeficientu.

12 lentelė. MobileNet tinklo šaligatvių įtrūkimų klasifikavimo sumaišymo matrica

	Su įtrūkimu	Be įtrūkimo
Su įtrūkimu	348	436
Be įtrūkimo	51	6463

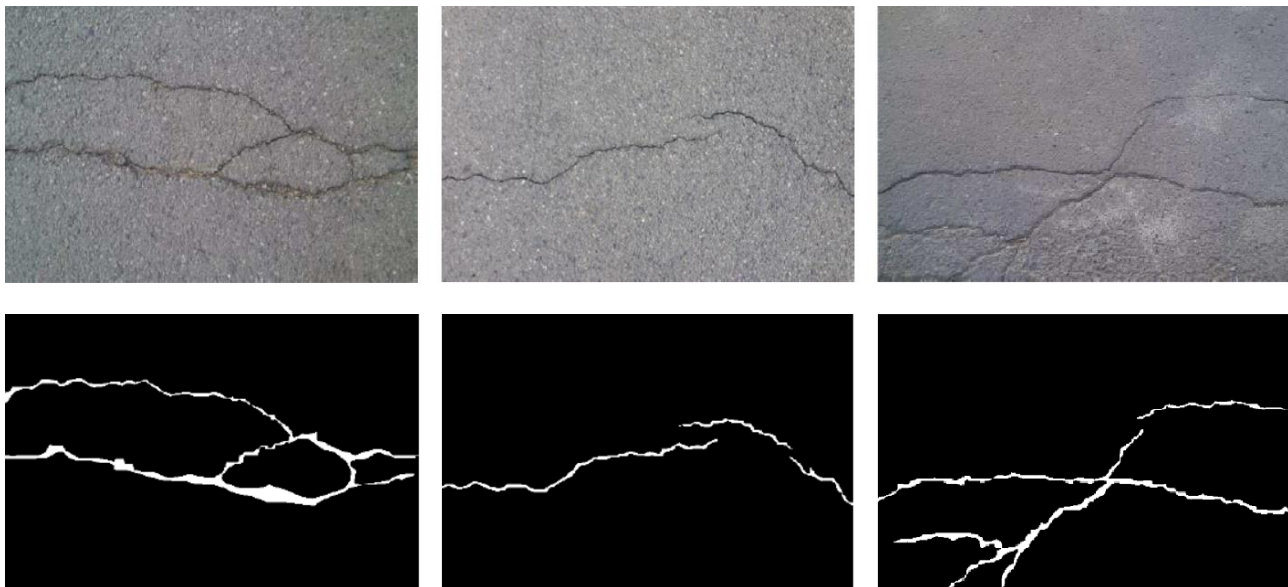


15 pav. MobileNet tinklo suklasifikuotų šaligatvio nuotraukų pavyzdys

- Išanalizavus tinklo klasifikuotas nuotraukas 15 pav., pastebėta, kad tinklas maišo šešėlius su įtrūkimais, bei stambią šaligatvio struktūrą priskiria įtrūkimams.

3.3. Defekto segmentavimas pikselio lygyje

Paskutinio tyrimo metu, defektų aptikimui pikselio lygyje, buvo apmokytas U-Net tipo tinklas. U-Net tinklo apmokymui, kitaip negu anksčiau aprašytuose eksperimentuose, reikalingos ne tik nuotraukos su įtrūkimais, bet ir jų požymių žemėlapiai. Kadangi naudota „SDNET2018“ duomenų bazė neturi savybių žemėlapių, buvo panaudota „CrackForest“ duomenų bazė. Minėta duomenų bazė yra sudaryta iš 118 paveikslėlių su įvairiais dangos įtrūkimais, kurie pasitelkiant augmentacijos metodus buvo padauginti iki 333 paveikslėlių, kiekvienam iš jų yra priskirtas ir individualus savybių žemėlapis, su vieta, kur konkrečiai įtrūkimas yra (žr 16 pav.).



16 pav. „CrackForest“ duomenų bazės paveikslėliai (viršuje) ir individualūs savybių žemėlapiai (apačioje)

Kadangi „SDNET2018“ duomenų bazė yra sudaryta iš 256x256 dydžio, o „CrackForest“ iš 480x320 dydžio vaizdų, pastarosios kiekvienas iš paveikslėlių buvo sukarpyti į 8 atskirus 256x256 dydžio paveikslėlius. Taip suvienodinant duomenų bazių paveikslėlių dydžius, pritaikant juos tinklo įėjimui ir augmentuojant pačią duomenų bazę iki 2664 paveikslėlių.

Tinklo jautrumą apibūdina filtro dydis, jis nurodo, koku žingsniu tinklas nuskaitys požymių žemėlapi, o paskutinėje stadijoje pačią nuotrauką tinklo įėjime.

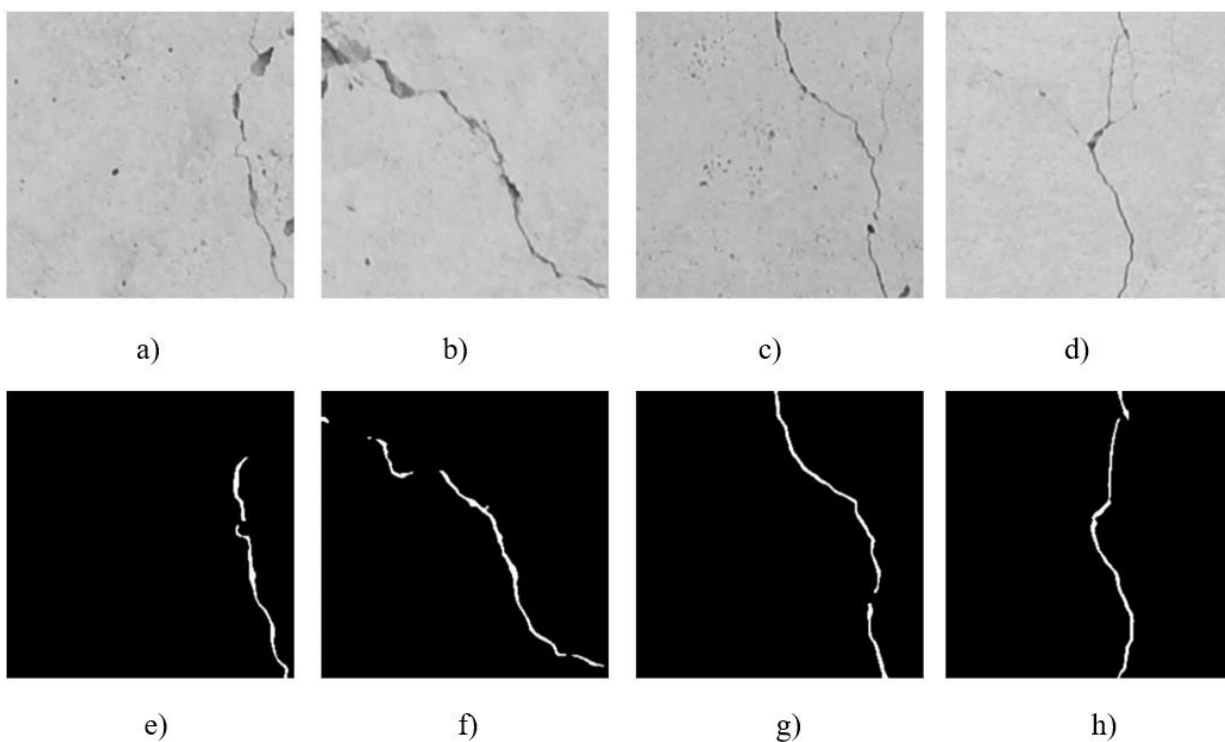
U-Net tinklo mokymui naudotos 4 tinklo modifikacijos. Kiekviena iš jų turėjo 80 iteracijų mokymą su 2344 paveikslėlių mokymo duomenų imtimi ir 320 paveikslėlių validavimo duomenų imtimi. Tolesnei tyrimo eigai, buvo naudotas geriausia – 0,645 F_1 įvertį įgijęs tinklas.

Į tinklo įėjimą buvo paduotos 24 nuotraukos:

- 4 nuotraukos su defektais sienoje;
- 4 nuotraukos be defektų sienoje;
- 4 nuotraukos su defektu kelio dangoje;
- 4 nuotraukos be defektų kelio dangoje;
- 4 nuotraukos su defektais ant šaligatvio tekstūros;
- 4 nuotraukos be defektų ant šaligatvio tekstūros.

3.3.1. Sienos segmentavimas

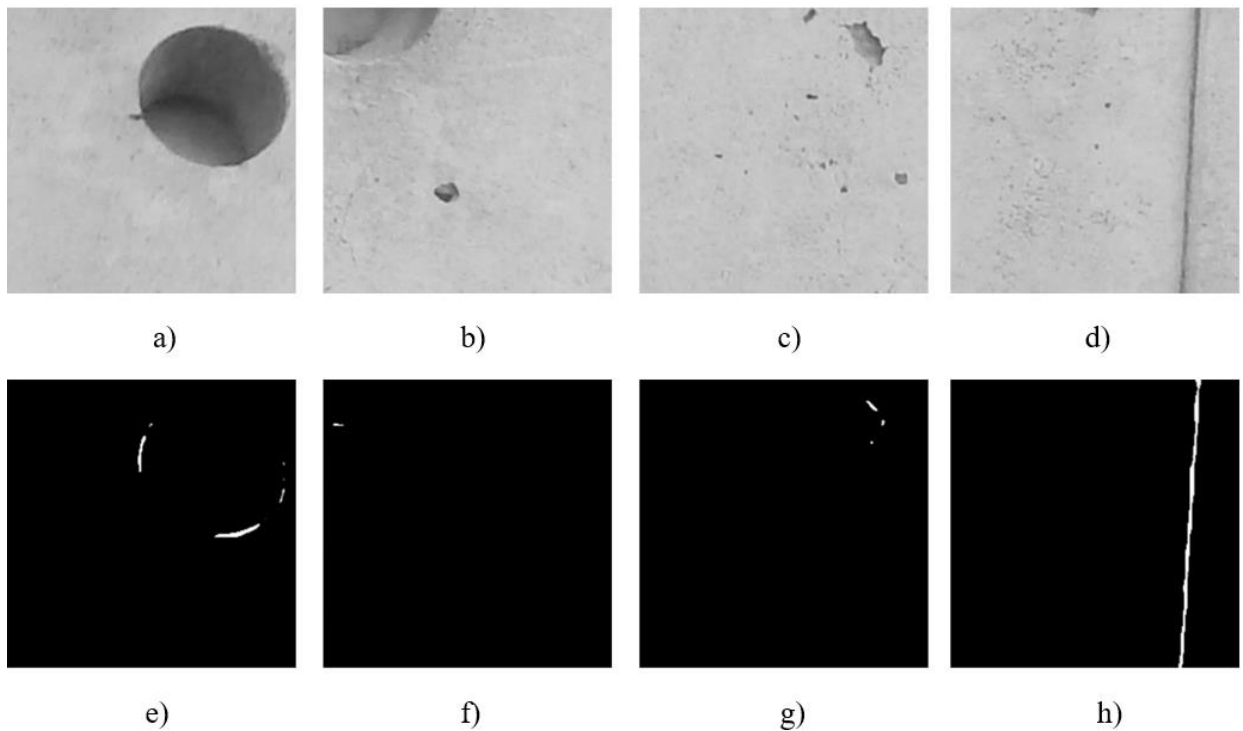
Į tinklo įėjimą padavus 4 nuotraukas (žr. 17 pav. a;b;c;d) su įtrūkimais sienose, tinklas dalinai juos atpažino ir išskyrė savo sudarytą segmentacijos žemėlapi (žr. 17 pav. e;f;g;h). Iš tinklo išėjimo matome, kad jis atpažįsta pagrindinį trūkį, tačiau neatpažįsta smulkių trūkio išsišakojimų. Taip gali būti dėl daugelio priežasčių, pagrindinės galėtų būti filtro dydis, riboti kompiuterio resursai, arba U-Net tinklo struktūra.



17 pav. Sienos paviršiaus segmentavimo rezultatai, kai sienoje yra defektas

Sekančiame etape buvo atrinktos 4 nuotraukos be įtrūkimų sienoje (žr. 18 pav. a;b;c;d). Tačiau, jos buvo parinktos tokios, kuriose yra konstrukcijoje numatyti objektai tokie kaip apvalios ertmės (žr. 18 pav. a) ar paviršiaus sujungimo siūlės (žr. 18 pav. d).

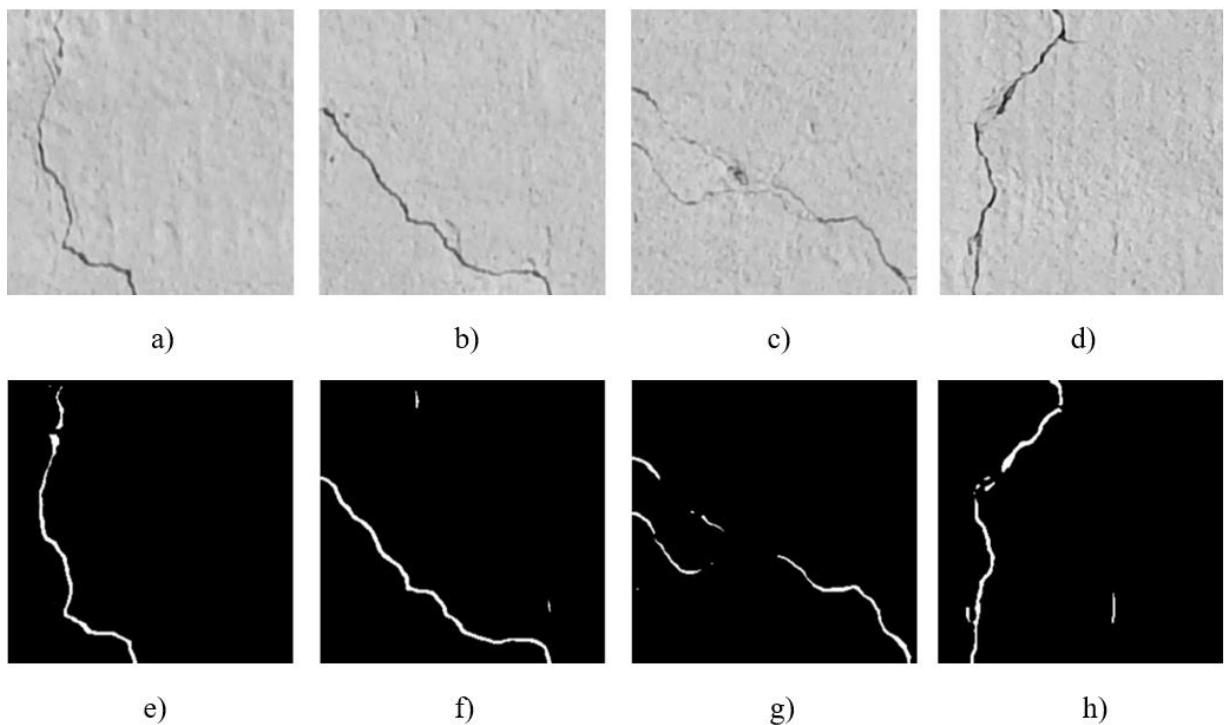
Iš tinklo išėjimo matome, kad kaip ir galvojama, tinklas dalinai pažymi vietas, kuriose nėra jokių defektų (žr. 18 pav. e), taip pat pažymi visą sujungimo siūlę (18 pav. h). Taip pat, tinklas pažymėjo ir dalį paviršiaus ištrupėjimo (žr. 18 pav. g), kuris galėtų būti aktualus tiriant tokius paviršius.



18 pav. Sienos paviršiaus segmentavimo rezultatai, kai sienoje nėra defekto

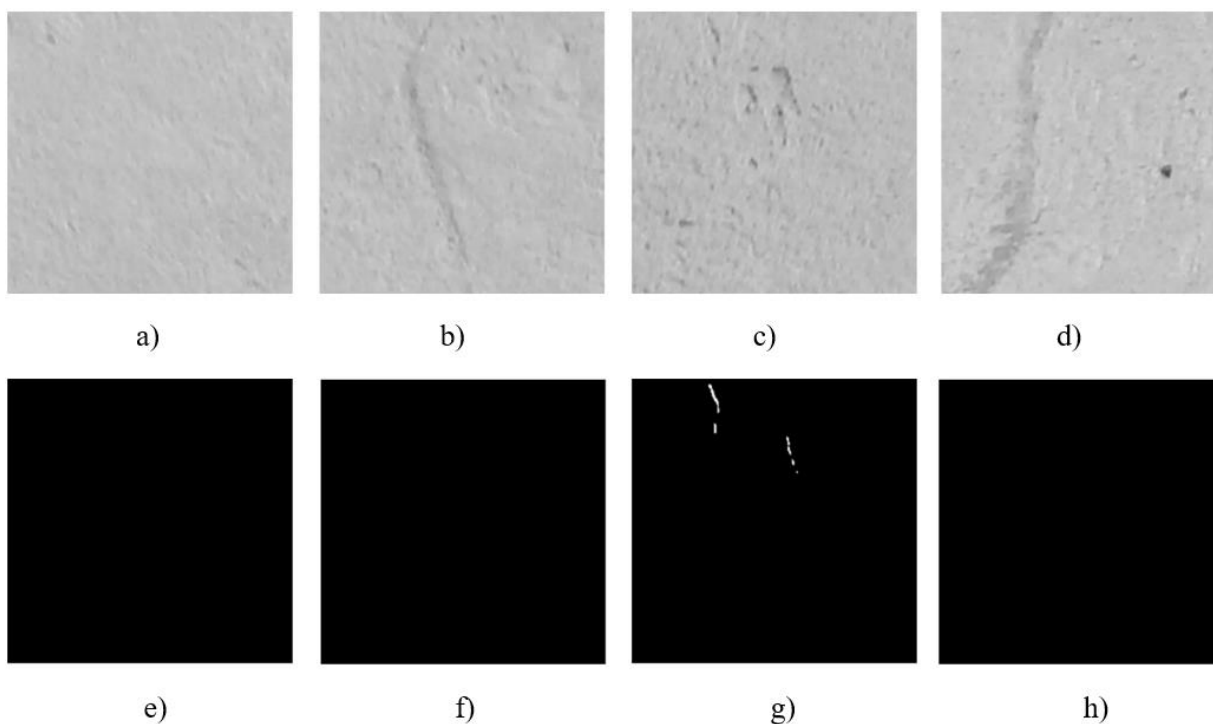
3.3.2. Kelio dangos segmentavimas

Analogiškai pirmam etapui, padaryta ir su kelio dangos nuotraukomis. Į tinklo įėjimą buvo paduota 4 nuotraukos su kelio dangos defektais (žr. 19 pav. a;b;c;d). Šios dangos atveju, matosi, kad tinklas geriau priėmė mokymą ir geriau įsidėmėjo defekto savybes. Kaip matome 19 paveikslo e,f,g ir h dalyse, tinklas atpažįsta gana menkus defekto išsišakojimus.



19 pav. Kelio dangos paviršiaus segmentavimo rezultatai, kai dangoje yra defektas

Tuomet patikrinimui, į tinklo įėjimą paduota 4 nuotraukos be kelio dangos defektų (žr. 20 pav. a;b;c;d). Tikėtasi, kad 20 nuotraukos b ir d dalyse matomi patamsėjimai, bent dalinai bus priskirti įtrūkimams, bet rezultatai geresni nei tikėtasi - segmentavimo žemėlapiuose matoma, kad tinklas tekstūros netobulumų netraktavo kaip defekto, tik 20 paveikslo c dalyje pastebėti keli ištrupėjimai (žr. 20 pav. g), kurie matomi ir akimi.

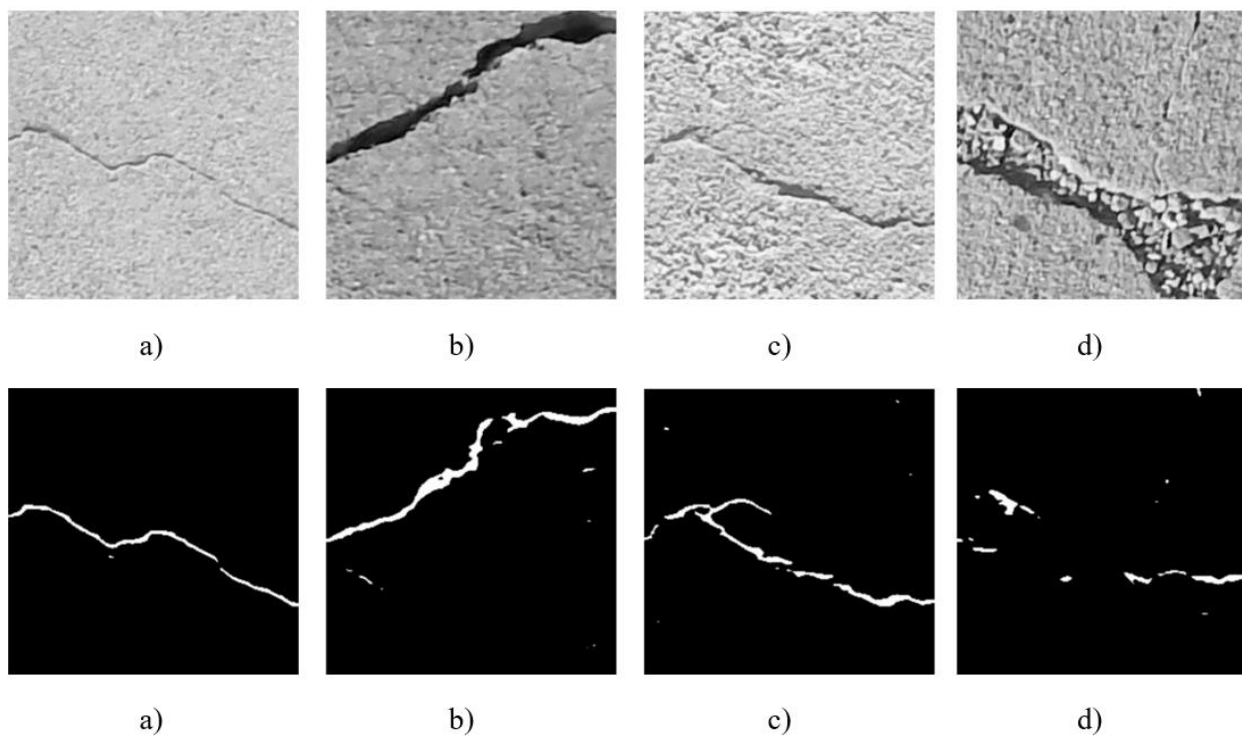


20 pav. Kelio dangos paviršiaus segmentavimo rezultatai, kai dangoje nėra defekto

3.3.3. Šaligatvio segmentavimas

Paskutinis tyrimas pakartotas su šaligatvio dangos nuotraukomis. Į tinklo įėjimą buvo paduotos 4 nuotraukos su defektais (žr. 21 pav. a;b;c;d). Nuotraukos buvo parinktos su skirtingo lygio defektais, kad būtų galima pamatyti tinklo galimybes.

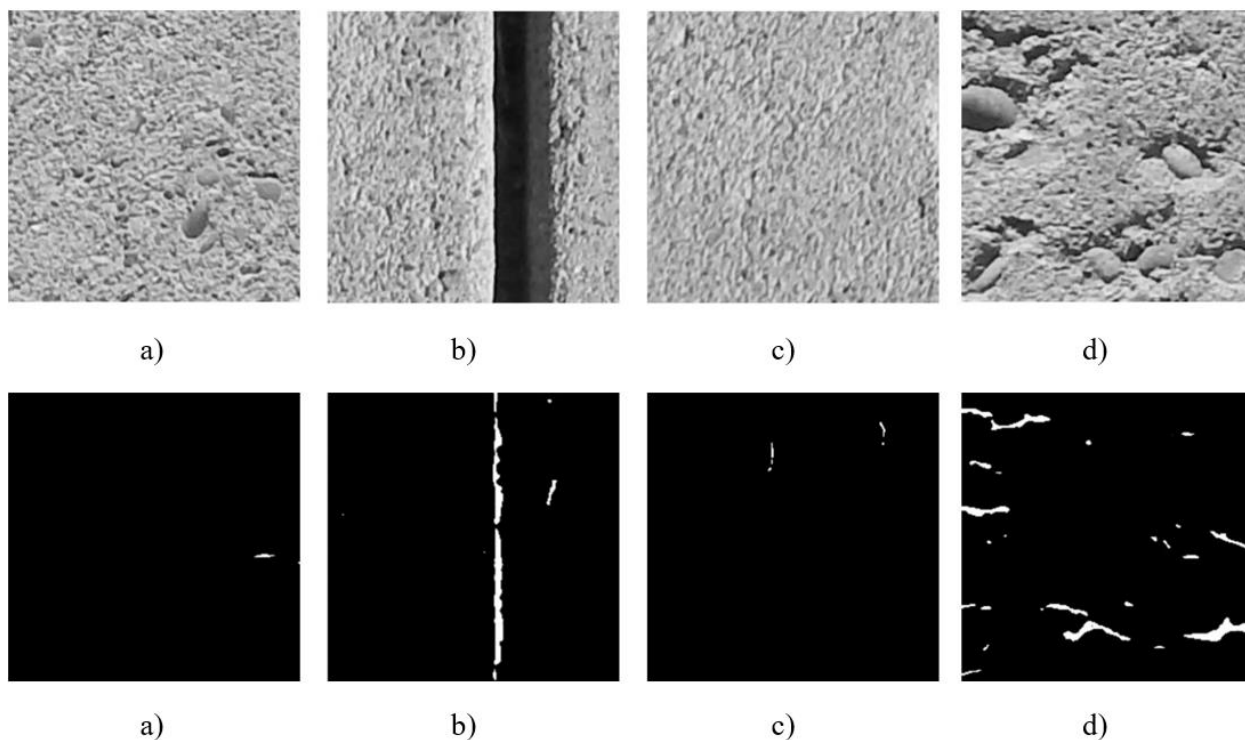
Tinklo išėjime matome, kad tinklas gana tiksliai atskiria ir nežymius (žr. 21 pav. d) ir susiliejusius su tekstūra (žr. 21 pav. g) įtrūkimus. Tačiau, 21 paveikslėlio h dalyje matome, kad didelį plotą užimantis ištrupėjimas buvo beveik neatpažintas, tai galėjo įvykti dėl stambios paviršiaus struktūros.



21 pav. Šaligatvio paviršiaus segmentavimo rezultatai, kai paviršiuje yra defektas

Norint pamatyti, kaip tinklas reaguoja į konstrukcijoje esančias kliūtis, tokias kaip dviejų paviršių susijungimai ar paviršiaus netolygumas, į tinklo įėjimą buvo paduotos 4 nuotraukos be defektų (žr. 22 pav. a;b;c;d).

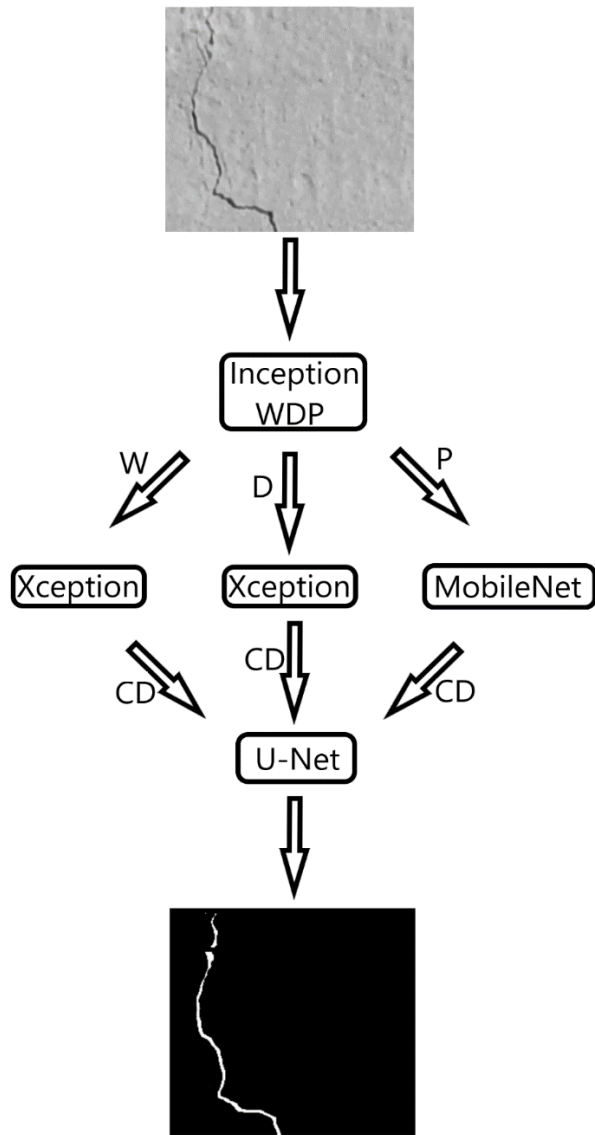
Iš tinklo išėjimo (žr. 22 pav. e;f;g;h) matoma, kad į minimalius netolygumus tinklas nereaguoja, tačiau 22 paveikslo f dalyje yra susegmentuotas dviejų paviršių sujungimas, o h dalyje yra stipriai sureaguota į didelius paviršiaus netolygumus.



22 pav. Šaligatvio paviršiaus segmentavimo rezultatai, kai paviršiuje nėra defekto

3.4. Eksperimentų apibendrinimas

Atlikus visus šiuos tyrimus, pastebėta, kad geriausias rezultatas turėtų būti pasiektas susisteminus visus šiuos eksperimentus. Kaip rodo tyrimų rezultatai, vienoje dangoje geriau defektus atpažįsta vienas tinklas, kitoje – kitas. Toliau pateikiamas siūlomas algoritmas (žr. 23 pav.), kurį vykdant, į tinklo įėjimą paduota nuotrauka pirmiausia klasifikuojama pagal dangą, tuomet, geriausiai rezultatus įgijęs tinklas nustato ar nuotraukoje yra defektas, ar jo nėra. Paskutiniame etape, jeigu nuotraukoje buvo atpažintas defektas, pasitelkiant U-Net tinklą jis yra segmentuojamas, taip naudotojui nurodant jo vietą ir formą.



23 pav. Sisteminis algoritmas (WDP – tekstūros atpažinimas; W – sienos; D – kelio danga; P – šaligatvis; CD – atpažintas defektas)

Išvados

1. Ištirus visus pasirinktus tinklus ir apžvelgus jų duotus rezultatus, geriausiai paviršiaus tekstūrą atskiria InceptionV3 tinklas.
2. InceptionV3 tinklas geriausiai atskiria šaligatvius, o labiausiai maišo kelio dangos ir sienų tekstūrą su šaligatviais.
3. Peržiūrėjus InceptionV3 atskirtas nuotraukas, ryškių skirtumų tarp gerai ir blogai klasifikuotų nuotraukų nepastebėta. Tikėtina, kad tinklas didesnę dėmesį skyrė šaligatvių požymiams, dėl to klaidingai priskiriama šiai grupei.
4. Ištirus pasirinktus tinklus ir apžvelgus rezultatus nustatyta, kad geriausiai su 0,88 F1 ir 0,42 MCC įverčiais, kelio dangos įtrūkimus klasifikuoja Xception tinklas.
5. Ištirus pasirinktus dirbtinius neuroninius tinklus nustatyta, kad sienų įtrūkimus geriausiai klasifikuoja Xception neuroninis tinklas, pasiekiantis 0,87 F1 įvertį su 0,56 MCC koeficientu.
6. Xception tinklas nėra toks jautrus, kad atpažinti nežymius sienų įtrūkimus. Taip pat, tinklas maišo žymesnius sienos tekstūros patamsėjimus su įtrūkimais.
7. Iš gautų rezultatų nustatyta, kad geriausiai šaligatvių paviršiaus įtrūkimus nustato MobileNet tinklas, kuris pasiekė 0,93 F1 įvertį su 0,59 MCC koeficientu.
8. Išanalizavus MobileNet tinklo išskirtas nuotraukas pastebėta, kad tinklas maišo šešėlius su įtrūkimais, bei stambią šaligatvio struktūrą priskiria įtrūkimams.
9. Išanalizavus U-Net tinklo išėjime gautas nuotraukas nustatyta, kad tinklas yra pajėgus segmentuoti ryškius įtrūkimus pikselio lygyje.
10. Nustatyta, kad U-Net tinklas nėra pakankamai jautrus aptikti pagrindinio įtrūkimo išsišakojimus.
11. Ištirus U-Net tinklo rezultatus, nustatyta, kad tinklas labai grubią paviršiaus struktūrą priskiria defektui.

Literatūros sąrašas

1. V. Jankauskas, J. Padgurskas, R. Rukuiža „Mašinų gamybos ir remonto technologijos“ Akademija, 2006m . Prieiga per internetą: <http://int.asu.lt/RRduomenys/MGRT-lab.pdf> [žiūrėta: 2019-12-12]
2. J. Malinowski „Kelių asfalto dangos plyšių didėjimo dinamika ir įtaka automobilių eismo parametrams“ 2011m. , Prieiga per internetą: <https://vb.vgtu.lt/object/elaba:1902833/index.html> [žiūrėta: 2019-12-12]
3. A. Rainys „Gaminio kokybės kontrolė taikant ultragaršą“ 2006 m. Prieiga per internetą: <http://gs.elaba.lt/object/elaba:2027093/2027093.pdf> [žiūrėta: 2019-12-12]
4. M. Gavilán, D. Balcones „Adaptive Road Crack Detection System by Pavement Classification“ 2011m., 11, 9628-9657, Prieiga per internetą: https://www.researchgate.net/publication/51873194_Adaptive_Road_Crack_Detection_System_by_Pavement_Classification [žiūrėta: 2019-03-11]
5. Z. Ebersold, N. Mitrovič, S. Dukič „Defectoscopy of Direct Laser Sintered Metals by LowTransmission Ultrasonic Frequencies“ Science of Sintering, 44, 2012 m. Prieiga per internetą: https://www.academia.edu/31660746/Defectoscopy_of_direct_laser_sintered_metals_by_low_transmission_ultrasonic_frequencies [žiūrėta: 2019-12-14]
6. S. Haykin „Neural Networks - A Comprehensive Foundation – Second Edition“, 2005m., Prieiga per internetą: http://amutiara.staff.gunadarma.ac.id/Downloads/files/14638/Neural_Networks._A_Compre_Found__2nd_.pdf [žiūrėta: 2019-12-12]
7. „Skaitmeninė patalogija“ 2016 m. , Prieiga per internetą: https://www.vpc.lt/skaitmenine-patalogija.html?fbclid=IwAR0IRSzE_6lwDVxQbzR5WyOxWoyW_isw4FxV27xsKo3OWO__Roh-Gmq8UEI [žiūrėta: 2019-12-12]
8. J. Dargienė „Austos struktūros deformacijų tyrimas vaizdų analizės metodu“ 2015 m. Prieiga per internetą: <https://epubl.ktu.edu/object/elaba:11680743/11680743.pdf> [žiūrėta: 2019-12-12]
9. B.K. Behera, T.E. Madan Mohan „Objective measurement of pilling by image processing technique“ 2005m. , Prieiga per internetą: https://www.emerald.com/insight/content/doi/10.1108/09556220510616156/full/html?casa_token=rrSIHgLk9cAAAAA:3hD1Wei2DKDTqbbqV2jt9Djo6buaAp3zc5MWoxESmsY_87QH4WP_r0LttbWm3gmD5lgU1VhTOU2UmjWGIMdkM5vqoFhoSsjuvR-K00j324nyQH9iYMRw [žiūrėta: 2019-12-18]
10. S. Haykin „Neural Networks - A Comprehensive Foundation – Second Edition“, 2005m., Prieiga per internetą: http://amutiara.staff.gunadarma.ac.id/Downloads/files/14638/Neural_Networks._A_Compre_Found__2nd_.pdf [žiūrėta: 2019-04-25]
11. I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press,163-166 2016m., Prieiga per internetą: http://thuvien.thanglong.edu.vn:8081/dspace/bitstream/DHTL_123456789/4227/1/10.4-1.pdf [žiūrėta: 2019-05-05]
12. Y. LeCun, Y. Bengio, G. Hinton „Deep learning“ ,“NATURE“, VOL 521,436 - 444, 2015m., Prieiga per internetą: https://creativecoding.soe.ucsc.edu/courses/cs523/slides/week3/DeepLearning_LeCun.pdf [žiūrėta: 2019-05-05]

13. A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," Curran Associates, Inc., 2012, pp. 1097-1105., Prieiga per internetą: <http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networks.pdf> [žiūrėta: 2019-05-16]
14. P. Kinderis „On-road vehicle recognition using neural networks and graphics processing unit“, 2011m. , Prieiga per internetą: <https://epublications.vu.lt/object/elaba:2155524/> [žiūrėta: 2019-04-22]
15. Y. LeCun, "Learning invariant feature hierarchies" in Computer Vision – ECCV 2012. Workshops and Demonstrations, 2012m. , pp. 496-505.
16. M. Gavilán, D. Balcones „Adaptive Road Crack Detection System by Pavement Classification“ 2011m., 11, 9628-9657, Prieiga per internetą: https://www.researchgate.net/publication/51873194_Adaptive_Road_Crack_Detection_System_by_Pavement_Classification [žiūrėta: 2019-04-16]
17. Y. LeCun, "Gradient-based learning applied to document recognition," in Proceedings of the IEEE 86.11, 1998m., Prieiga per internetą: <http://yann.lecun.com/exdb/publis/pdf/lecun-01a.pdf> [žiūrėta: 2019-05-11]
18. Ronneberger, O “U-Net: Convolutional Networks for Biomedical Image Segmentation”, 2015m. ,Prieiga per internetą: [https://arxiv.org/pdf/1505.04597.pdf%E5%92%8C%5bTiramisu%5d\(https://arxiv.org/abs/1611.09326.pdf](https://arxiv.org/pdf/1505.04597.pdf%E5%92%8C%5bTiramisu%5d(https://arxiv.org/abs/1611.09326.pdf) [žiūrėta: 2019-05-11]
19. Y. LeCun, " Road Extraction by Deep Residual U-Net" IEEE, Vol. 15,NO.5, 2018m. , Prieiga per internetą: <https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=8309343> (žiūrėta: 2019-05-18)
20. H. Qassim, A. Verma, D. Feinzimer „Compressed Residual-VGG16 CNN Model for Big Data Places Image Recognition“ 2018 m. , Prieiga per internetą: <https://vermaabhi23.github.io/publication/2018CCWC.pdf> [žiūrėta: 2019-12-18]
21. Muneeb ul Hassan „VGG16 – Convolutional Network for Classification and Detection“ 2018 m. , Prieiga per internetą: <https://neurohive.io/en/popular-networks/vgg16/> [žiūrėta: 2019-12-18]
22. „VGG19“ Prieiga per internetą: <https://keras.io/applications/#vgg19> [žiūrėta: 2019-12-18]
23. „VGG Net“ Prieiga per internetą: https://en.everybodywiki.com/VGG_Net [žiūrėta: 2019-12-18]
24. K. Simonyan, A. Zisserman „Very deep convolutional networks for large-scale image recognition“ 2015 m. , Prieiga per internetą: <https://www.robots.ox.ac.uk/~vgg/publications/2015/Simonyan15/simonyan15.pdf> [žiūrėta: 2019-12-20]
25. C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed „Going deeper with convolutions“ 2015 m. , Prieiga per internetą: <https://ieeexplore.ieee.org/ielx7/7293313/7298593/07298594.pdf?tp=&arnumber=7298594&isnumber=7298593&ref=aHR0cHM6Ly9pZWVleHBsb3JlLmllZWUub3JnL2RvY3VtZW50LzcyOTg1OTQ=> [žiūrėta: 2019-12-20]
26. C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens „Rethinking the Inception Architecture for Computer Vision“ 2016 m. , Prieiga per internetą: https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/Szegedy_Rethinking_the_Inception_CVP_R_2016_paper.pdf [žiūrėta: 2019-12-20]
27. Md. R. B. Wahed „Comparative Analysis between Inception-v3 and Other Learning Systems using Facial Expressions Detection“ 2016 m. , Prieiga per internetą: https://pdfs.semanticscholar.org/7e27/53f48bccf09c95706f9f0cd9fea59544518e.pdf?_ga=2.179054628.1687375119.1579386110-1833545081.1578411591 [žiūrėta: 2019-12-20]

28. „Pruning depthwise separable convolutions for extra efficiency gain of lightweight models“ 2020 m. , Prieiga per internetą: https://openreview.net/attachment?id=B1g5qyHYPS&name=original_pdf [žiūrėta: 2019-12-20]
29. A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko „MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications“ 2017 m. , Prieiga per internetą: <https://arxiv.org/pdf/1704.04861.pdf> [žiūrėta: 2020-01-06]
30. F. Wan „Deep Learning Method used in Skin Lesions Segmentation and Classification“ 2018 m. , Prieiga per internetą: <https://pdfs.semanticscholar.org/4928/2e58d454f295b30e753d280717e12894dc87.pdf> [žiūrėta: 2020-01-06]
31. I. S. Jayasinghe, C. L. Senanayake, S. I. M. Kahawandala „Video Classification using Pre-trained models in the Convolutional Neural Networks“ 2016 m. , Prieiga per internetą: http://project.sliit.lk/redmine/attachments/download/3696/research_paper%20%2018-19J%2011.pdf [žiūrėta: 2020-01-06]
32. „Aerial Image Semantic Segmentation using Neural Search Network architecture“ 2018 m. , Prieiga per internetą: https://eprints.uet.vnu.edu.vn/eprints/id/eprint/3264/1/Thin_miwai_0703_paper.pdf [žiūrėta: 2020-01-10]
33. R. Marwaha, B. Fataniya „Classification of Indian Herbal Plants based on powder microscopic images using Transfer Learning“ 2018 m. , Prieiga per internetą: <https://ieeexplore.ieee.org/iel7/8735657/8745714/08745922.pdf> [žiūrėta: 2020-01-10]
34. H. Xu, S. Park, S. H. Lee, T. H. Hwang „Using transfer learning on whole slide images to predict tumor mutational burden in bladder cancer patients“ 2019 m. , Prieiga per internetą: <https://www.biorxiv.org/content/biorxiv/early/2019/02/19/554527.full.pdf> [žiūrėta: 2020-01-10]
35. S. V. Stehman, “Selecting and Interpreting Measures of Thematic Classification Accuracy”, vol.62, 77–89, 1997m., Prieiga per internetą: https://www.researchgate.net/profile/Stephen_Stehman/publication/222169047_Selecting_andinterpreting_measures_of_thematic_classification_accuracy/links/5b5a0fe5a6fdccf0b2f8fe87/Selecting-and-interpreting-measures-of-thematic-classification-accuracy.pdf [žiūrėta: 2019-05-18]
36. D. M. W. Powers, “Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation”, 37-63, 2011m., Prieiga per internetą: <https://dspace2.flinders.edu.au/xmlui/bitstream/handle/2328/27165/Powers%20Evaluation.pdf?sequence=1&isAllowed=y> [žiūrėta: 2019-05-19]
37. „Jaccard index“ Prieiga per internetą: https://en.wikipedia.org/wiki/Jaccard_index [žiūrėta: 2019-05-22]
38. S. Dorafshan, R. J. Thomas, M. Maguire „SDNET2018: An annotated image dataset for non-contact concrete crack detection using deep convolutional neural networks“ 2018 m. , Prieiga per internetą: https://digitalcommons.usu.edu/cgi/viewcontent.cgi?article=4611&context=cee_facpub [žiūrėta: 2020-01-12]
39. C. Shorten, T. M. „Khoshgoftaar „A survey on Image Data Augmentation for Deep Learning“ 2019 m. , Prieiga per internetą: <https://link.springer.com/article/10.1186/s40537-019-0197-0> [žiūrėta: 2020-01-12]
40. J. Wang, L. Perez, „The Effectiveness of Data Augmentation in Image Classification using Deep Learning“ 2017 m. , Prieiga per internetą: <http://cs231n.stanford.edu/reports/2017/pdfs/300.pdf> [žiūrėta: 2020-01-12]
41. „Matthews correlation coefficient“ Prieiga per internetą: https://en.wikipedia.org/wiki/Matthews_correlation_coefficient [žiūrėta: 2019-05-22]