



**Kauno technologijos universitetas**  
Matematikos ir gamtos mokslų fakultetas

**Giliojo mokymosi metodai betono įtrūkimams aptikti**  
Baigiamasis magistro studijų projektas

---

**Gedmantas Čepkauskas**  
Projekto autorius

**doc. dr. Tomas Iešmantas**  
Vadovas

---

**Kaunas, 2022**



**Kauno technologijos universitetas**

Matematikos ir gamtos mokslų fakultetas

## **Giliojo mokymosi metodai betono įtrūkimams aptikti**

Baigiamasis magistro studijų projektas

Taikomoji matematika (6211AX006)

---

**Gedmantas Čepkauskas**

Projekto autorius

(parašas)

(data)

**doc. dr. Tomas Iešmantas**

Vadovas

(parašas)

(data)

**doc. dr. Paulius Palevičius**

Recenzentas

(parašas)

(data)

---

**Kaunas, 2022**



**Kauno technologijos universitetas**

Matematikos ir gamtos mokslų fakultetas

Gedmantas Čepkauskas

## **Giliojo mokymosi metodai betono įtrūkimams aptikti**

Akademinio sąžiningumo deklaracija

Patvirtinu, kad:

1. baigiamąjį projektą parengiau savarankiškai ir sąžiningai, nepažeisdama(s) kitų asmenų autoriaus ar kitų teisių, laikydamasi(s) Lietuvos Respublikos autorių teisių ir gretutinių teisių įstatymo nuostatų, Kauno technologijos universiteto (toliau – Universitetas) intelektinės nuosavybės valdymo ir perdavimo nuostatų bei Universiteto akademinės etikos kodekse nustatytų etikos reikalavimų;
2. baigiamajame projekte visi pateikti duomenys ir tyrimų rezultatai yra teisingi ir gauti teisėtai, nei viena šio projekto dalis nėra plagijuota nuo jokių spausdintinių ar elektroninių šaltinių, visos baigiamojo projekto tekste pateiktos citatos ir nuorodos yra nurodytos literatūros sąrašė;
3. įstatymų nenumatytų piniginių sumų už baigiamąjį projektą ar jo dalis niekam nesu mokėjęs (-usi);
4. suprantu, kad išaiškėjus nesąžiningumo ar kitų asmenų teisių pažeidimo faktui, man bus taikomos akademinės nuobaudos pagal Universitete galiojančią tvarką ir būsiu pašalinta(s) iš Universiteto, o baigiamasis projektas gali būti pateiktas Akademinės etikos ir procedūrų kontrolieriaus tarnybai nagrinėjant galimą akademinės etikos pažeidimą.

Gedmantas Čepkauskas

*Patvirtinta elektroniniu būdu*

Gedmantas Čepkauskas. Giliojo mokymosi metodai betono įtrūkimams aptikti. Magistro studijų baigiamasis projektas / vadovas doc. dr. Tomas Iešmantas; Kauno technologijos universitetas, Matematikos ir gamtos mokslų fakultetas.

Studijų kryptis ir sritis (studijų krypčių grupė): Taikomoji matematika (Matematikos mokslai).

Reikšminiai žodžiai: vaizdų analizė, įtrūkimas, gilieji neuroniniai tinklai, EfficientNet, U-Net.

Kaunas, 2022. 43 p.

## Santrauka

Magistro baigiamajame darbe nagrinėjama problematika, kaip naudojant skirtingas dirbtinių neuroninių tinklų architektūras klasifikuoti betono vaizdus į sveikus ir turinčius įtrūkimų, o vėliau įvairiais būdais atlikti įtrūkimų segmentavimą.

Metodinėje dalyje aprašomas sąsukų neuroninio tinklo veikimo principas, kaip atliekamas tinklo apmokymas, nupasakojant stochastinio gradientinio nusileido ir ADAM optimizatoriaus svarbą. Vėliau apžvelgiamos skirtingos sąsukų neuroninių tinklų architektūros, kurios naudotos tyrime. Atskirai dėmesys skiriamas U-Net tipo architektūrai, kurios pagalba atliekamas semantinis įtrūkimų segmentavimas.

Atliekant klasifikavimo užduotį nagrinėti EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB3 modeliai, apmokymo metu tiek naudojant, tiek nenaudojant vaizdų augmentaciją. Vėliau ieškant geriausio klasifikatoriaus išbandyti ir AlexNet bei VGG-16 modeliai, be vaizdų augmentacijos. Visi modeliai apmokyti naudojantis „Surface Crack Detection“ vaizdų rinkiniu, kurį sudaro 40000 skirtingų cheminių savybių ir atspalvių betono vaizdų, iš jų pusė vaizduoja įtrūkimą, pusė sveiką betoną. Po apmokymo kiekvienas modelis išbandytas su 1000 aukštos raiškos įtrūkusio betono vaizdų. Vertinant kiekvieno modelio tikslumą testavimo imtyje, gauti labai panašūs rezultatai, todėl nuspręsta geriausią modelį išrinkti apskaičiuojant kokią dalį aukštos raiškos vaizdų modelis klasifikuoja teisingai. Šią užduotį geriausiai atliko EfficientNetB3 modelis, apmokytas nenaudojant vaizdų augmentacijos. Jis teisingai klasifikavo 952 vaizdus iš 1000.

Kitu etapu atliktas grubus vaizdo klasifikavimas. Jo metu,  $1024 \times 1024$  formato nuotrauka skaidoma į  $64 \times 128 \times 128$  dydžio zonas ir kiekvienoje iš jų, naudojant jau apmokytą klasifikavimo modelį, vertinama tikimybė, kad yra įtrūkimas, o rezultatas atvaizduojamas karščio žemėlapiu. Atliekant šią užduotį lyginti rezultatai gauti geriausiu klasifikavimo modeliu EfficientNetB3, be vaizdų augmentacijos, ir modelių ansambliu, sudarytu iš EfficientNetB3 ir EfficientNetB2, be vaizdų augmentacijos ir EfficientNetB1 su vaizdų augmentacija. Modelių ansamblis nepateikė reikšmingai geresnių rezultatų, todėl grubiam klasifikavimui siūloma naudoti tik EfficientNetB3 modelį, apmokytą be vaizdų augmentacijos.

Vėliau semantiniam įtrūkimų segmentavimui išbandytos U-Net architektūros modifikacijos, su skirtingais pagrindiniais modeliais: EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB3, EfficientNetB4 ir EfficientNetB5. Geriausiai semantinį segmentavimą atliko U-Net architektūra su EfficientNetB0 pagrindiniu modeliu.

Čepkauskas Gedmantas. Deep learning methods for concrete crack detection / supervisor doc. dr. Tomas Iešmantas; Faculty of Mathematics and Natural Sciences, Kaunas University of Technology.

Study field and area (study field group): Applied Mathematics (Mathematical Sciences).

Keywords: image analysis, crack, deep neural networks, EfficientNet, U-Net.

Kaunas, 2022. 43 p.

### **Summary**

The master's thesis deals with the problem of how to classify concrete images into healthy and cracked ones by using different artificial neural network architectures, and then to perform crack segmentation in different ways.

The methodological part describes the principle of operation of the convolutional neural network, how the network training is performed, explaining the importance of the stochastic gradient descent and the ADAM optimizer. The different architecture of the convolutional neural networks used in the study are then reviewed. Special attention is paid to the U-Net type architecture, which is used for semantic crack segmentation.

During the classification task, the models EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB3 were examined, with and without the use of image augmentation during the training. Later, AlexNet and VGG-16 models without image augmentation were tested to find the best classifier. All models are trained using the "Surface Crack Detection" image set, which consists of 40000 images of concrete with different chemical properties and shades, half of which have a crack and half of which is intact concrete. After training, each model was tested with 1000 images of high-resolution cracked concrete. Assessing the accuracy of each model in the test sample yielded very similar results, so it was decided to select the best model by calculating what proportion of the high-resolution image model classifies correctly. This task was best performed by the EfficientNetB3 model trained without the use of image augmentation. It correctly classified 952 images of 1000.

In the next step, a rough image segmentation was performed. During it, a  $1024 \times 1024$  image is divided into  $64 \times 128 \times 128$  size tiles, and in each of them, using an already trained classification model, the probability of a crack is assessed and the result is displayed in the form of a heat map. The results of this task were compared using the best classification model EfficientNetB3, without image augmentation, and the ensemble of models consisting of EfficientNetB3 and EfficientNetB2, without image augmentation, and EfficientNetB1 with image augmentation. The model ensemble did not provide significantly better results, so the EfficientNetB3 model trained without image augmentation is proposed for a rough classification.

Subsequently, different types of the U-Net architecture were tested for semantic crack segmentation, with different backbone models: EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB3, EfficientNetB4, and EfficientNetB5. The U-Net architecture with the EfficientNetB0 backbone model performed the best.

## Turinys

<b>Lentelių sąrašas.....</b>	<b>7</b>
<b>Paveikslų sąrašas .....</b>	<b>8</b>
<b>Santrumpų ir terminų sąrašas.....</b>	<b>9</b>
<b>Įvadas.....</b>	<b>10</b>
<b>1. Literatūros apžvalga .....</b>	<b>11</b>
<b>2. Duomenys ir tyrimo metodai.....</b>	<b>19</b>
2.1. Duomenys.....	19
2.2. Tyrimo metodai .....	20
2.2.1. CNN.....	21
2.2.2. Architektūros .....	23
2.2.3. Tyrimo eiga .....	27
<b>3. Tyrimų rezultatai ir jų aptarimas.....</b>	<b>29</b>
3.1. Klasifikavimas.....	29
3.1.1. EfficientNetB3 be augmentacijos.....	31
3.1.2. EfficientNetB2 be augmentacijos.....	32
3.1.3. EfficientNetB1 su augmentacija.....	33
3.2. Segmentavimas.....	33
3.2.1. Segmentavimas su EfficientNetB3, be vaizdų augmentacijos .....	34
3.2.2. Segmentavimas su modelių ansambliu.....	34
3.2.3. Segmentavimo su EfficientNetB3, be vaizdų augmentacijos, modelio ir ansamblio rezultatų palyginimas.....	35
3.2.4. Segmentavimas naudojant U-Net .....	36
<b>Išvados .....</b>	<b>39</b>
<b>Literatūros sąrašas .....</b>	<b>40</b>

## Lentelių sąrašas

<b>1 lentelė.</b> U-Net ir CrackSegNet modelių palyginimai [23] .....	15
<b>2 lentelė.</b> Skirtingų modelių detalės ir tikslumas [41].....	18
<b>3 lentelė.</b> Klasifikavimo modelių rezultatai .....	29
<b>4 lentelė.</b> EfficientNetB3 be augmentacijos modelio klasifikavimo rezultatai .....	31
<b>5 lentelė.</b> EfficientNetB3 be augmentacijos modelio klasifikavimo rezultatai .....	32
<b>6 lentelė.</b> EfficientNetB1 su augmentacija modelio klasifikavimo rezultatai .....	33
<b>7 lentelė.</b> U-Net rezultatai .....	36

## Paveikslų sąrašas

1 pav. Originali nuotrauka ir joje rasti įtrūkimai [11] .....	11
2 pav. Semantinio segmentavimo rezultatai [17] .....	13
3 pav. Aprėpties langai apie įtrūkimą ir lazerių taškus [19] .....	14
4 pav. AE signalų rezultatai [33] .....	17
5 pav. Įtrūkusio a ir sveiko b betono pavyzdys iš apmokymo imties .....	19
6 pav. Didelės raiškos betono su įtrūkimu pavyzdys .....	20
7 pav. Įtrūkimo a ir jo kaukės b pavyzdys .....	20
8 pav. Sąsūkos operacija [44] .....	21
9 pav. Aktyvacijos funkcijos [45] .....	21
10 pav. Maksimalaus telkimo pavyzdys [46] .....	21
11 pav. Sąsūkų neuroninis tinklas [47] .....	22
12 pav. EfficientNet architektūros schema [51] .....	24
13 pav. VGG-16 tinklo architektūra [56] .....	25
14 pav. AlexNet architektūros schema [58] .....	26
15 pav. U-Net architektūra [59] .....	27
16 pav. Klasifikavimo modelių rezultatai .....	30
17 pav. Sutrupėjusio betono pavyzdys .....	31
18 pav. EfficientNetB3 be augmentacijos modelio tikslumo a ir nuostolių funkcijos reikšmės b kitimas, validacijos imtyje, apmokymo metu .....	32
19 pav. EfficientNetB2 be augmentacijos modelio tikslumo a ir nuostolių funkcijos reikšmės b kitimas, validacijos imtyje, apmokymo metu .....	32
20 pav. EfficientNetB1 su augmentacija modelio tikslumo a ir nuostolių funkcijos reikšmės b kitimas, validacijos imtyje, apmokymo metu .....	33
21 pav. Segmentuotų vaizdų, su EfficientNetB3 modeliu, be vaizdų augmentacijos, pavyzdys .....	34
22 pav. Segmentuotų vaizdų, su modelių ansambliu, pavyzdys .....	35
23 pav. Segmentavimo rezultatai su EfficientNetB3, be vaizdų augmentacijos ir su modelių ansambliu .....	35
24 pav. U-Net rezultatai .....	36
25 pav. Segmentavimo su U-Net pavyzdžiai .....	38



## Santrumpų ir terminų sąrašas

### Santrumpos:

ANN – dirbtinis neuroninis tinklas (angl. *artificial neural network*)

RNN – rekurentinis neuroninis tinklas (angl. *recurrent neural network*)

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

JAV – Jungtinės Amerikos Valstijos

RGB – spalvotas vaizdas (angl. *red green blue*)

IoU - tikslumo matas (angl. *Intersection over Union*)

## Įvadas

Laikoma, jog pirmieji betoną pradėjo naudoti senovės graikai, apie 1400-sius metus prieš mūsų erą. Dėl tuo metu itin brangaus ir sudėtingo paruošimo, betonas naudotas tik liejant didikų rūmų grindis [1, 2]. Gerokai vėliau, apie 300-siais metais prieš Kristų, betono technologiją perėmė romėnai. Jų betonas pasižymėjo ilgaamžiškumu, buvo pigesnis, dėl ko ir naudotas gerokai plačiau [3]. Romėniškas betonas (lot. *opus caementicium*) gamintas iš negesintų kalkių, pemzos ir pucolanų, silpnai susicementavusių magminių ir nuosėdinių uolėnų, randamų Italijos Pozzuoli miesto apylinkėse. Laikoma, kad būtent kiek patobulinta betono formulė paskatino Romos architektūros proveržį, kurio geriausias pavyzdys Panteonas, su iki šiol didžiausiu, ne gelžbetoniniu kupolu pasaulyje [4,5]. Šiais laikais betonas naudojamas itin plačiai: gyvenamiesiems ir visuomeniniams pastatams, keliams, vamzdynams, tiltams, užtvankoms ir pan [6]. Visgi, tiek prieš 3000-čius metų, tiek dabar, betonas per laiką susidėvi. Jeigu jis nėra fiziškai gniuždomas, tada įtrūkimai betone atsiranda dėl to, jog mišinyje naudota per daug vandens ir jam išgaravus viduje atsiranda daug mikro įtrūkimų, kurie vėliau virsta didesniais plyšiais. Taipogi dėl atmosferinio poveikio, kada keičiantis temperatūrai, natūraliai esančiuose mikroskopiniuose įtrūkimuose besikaupiantis vanduo plečiasi ir traukiasi [7]. Vėliau įtrūkimai ima plėstis vis greičiau, kol galiausiai betoninė konstrukcija ima trupėti arba skyla. Tai gali turėti katastrofinių pasekmių, kaip nutiko Floridoje, Jungtinėse Amerikos Valstijose 2021 m. Per šią nelaimę, sugriuvus daliai gyvenamojo namo, žuvo 18 žmonių, keliolika sužeista. Apie galimą pavojų pranešimų sulaukta dar 2018 metais, kada pastebėti įtrūkimai sienose, tačiau jokių veiksmų nesiimta [8]. Šis pavyzdys tik parodo, jog iš pažiūros menki įtrūkimai vėliau gali turėti tragiškas pasekmes. Visgi dažniausiai būna pakankamai laiko užkirsti kelią tragedijoms. Tą padaryti galima laikantis betono gamybos technologinių reikalavimų, o vėliau reguliariai inspektuojant objektą, tačiau tai pakankamai brangu. Vykdamas objekto techninę priežiūrą, specialistai turi vykti į reikiamą vietą, atlikti detalią apžiūrą. Magistro baigiamojo darbo tikslas, sukurti modelį, grįstą giliojo mokymosi metodais, kuris aptiktų įtrūkimus betono konstrukcijose. Modelis gebėtų nustatyti ar pateiktoje betono nuotraukoje yra įtrūkimas, jei taip, vaizde pažymėti įtrūkimus keliais skirtingais būdais. Pritaikius šį sprendimą praktiškai, galima sumažinti tiltų, gyvenamųjų namų, kelių ir ypač vamzdynų inspektavimo kaštus. Tai leistų dažniau ir detaliau vykdyti šių objektų techninę priežiūrą, taip mažinant nelaimingų atsitikimų riziką. Tiesioginę naudą pajustų statybų ir transporto sektoriai dėl sumažėjusių infrastruktūros palaikymo kaštų. Ne tiesioginę naudą gali pajusti visi, žinodami, jog naudojama infrastruktūra nuolatos prevenciškai stebima ir analizuojama [9].

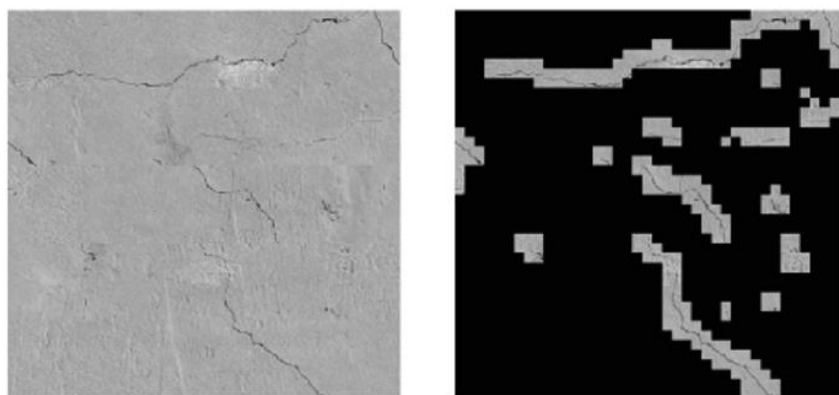
**Darbo tikslas:** Sudaryti ir iširti dirbtinių neuronų tinklais grįstą metodą, įtrūkimams betone aptikti ir segmentuoti.

### Darbo uždaviniai.

1. Apžvelgti metodus naudojamus įtrūkimų automatizuotai detekcijai betone.
2. Iširti pasirinktų architektūrų efektyvumą betono skaitmeninių vaizdų klasifikavimui.
3. Sudaryti ir iširti algoritmą, skirtą betono trūkių segmentavimui naudojant vien tik klasifikatorių.
4. Pritaikyti U-Net tipo architektūrą detaliam trūkių segmentavimui ir palyginti su 3 uždavinyje gautais rezultatais.

## 1. Literatūros apžvalga

Qianyun Zhang ir kt. 2020 metais publikuotame straipsnyje nagrinėja kaip automatizuoti įtrūkimų paiešką betoninėse konstrukcijose [11]. Minima, jog pagal Jungtinių Amerikos Valstijų civilinės inžinerijos departamento pranešimą, JAV yra 56000 struktūriškai skirtingų tiltų, kuriuos nuolatos reikia prižiūrėti [10]. Autoriai ieškodami sprendimo kaip automatizuoti objektų techninę priežiūrą, nusprendė naudoti 1 dimensijos sąsūkų neuroninius tinklus (angl. *one-dimensional convolutional neural network, 1D-CNN*) ir ilgos-trumpos atminties metodą (angl. *long short-term memory, LSTM*). 1D-CNN-LSTM modelio apmokymui naudota 4800 rankiniu būdu pažymėtų įtrūkusių ir neįtrūkusių nuotraukų, darytų betoninių tiltų perdangose. Pasitaikė įvairaus dydžio įtrūkimų, nuo 1 mm iki 15 mm pločio. Vaizdų dydis –  $256 \times 256$ . Siekiant pagerinti aptikimo tikslumą, nuotraukos suskirstytos į antrinius vaizdus su  $64 \times 64$  pikseliais. Iš 4800 vaizdų, 4300 apkarpomė į 17 200 mažo formato nuotraukas. Vaizdai, kurie yra neryškūs arba kuriuose yra kampų įtrūkimų pašalinami. Galiausiai 16 789 vaizdai saugomi kaip duomenų rinkiniai. Likę 500 tilto denio nuotraukų sujungtos į 20 vaizdų, kurių dydis  $1280 \times 1280$  pikselių, kad išbandytų sukurto klasifikatoriaus apibendrinimo pajėgumą. 1D-CNN-LSTM modelis sudarytas iš įvesties sluoksnio, keturių konvoliucinių sluoksnių, po kurių seka „maksimalus telkimas“ (angl. *max pooling*), LSTM, du pilnai sujungti sluoksniai bei išvestis. Apmokant 16789 vaizdai padalinti atitinkamai: 70% apmokymo imtis, 15% validacijos imtis ir 15% testavimo imtis. Su straipsnio autorių turima įranga apmokymas truko 1 valandą, 12 minučių. Pavyko pasiekti 99,05%, 98,9% ir 99,25% tikslumą atitinkamai apmokymo, validacijos ir testavimo imtyse. Vėliau geriausias modelis išbandytas su didelio formato,  $1280 \times 1280$ , vaizdais, kurie nebuvo naudoti iki tol (žr. 1 pav.). Modelis šiose nuotraukose įtrūkimus rasdavo per 5 – 7 sekundes. Palyginimui, su tais pačiais vaizdais standartinio 2D-CNN modelio apmokymas truko 2 valandas 45 minutes, naujose nuotraukose įtrūkimus aptikdavo per 38 – 59 sekundes, naudojant tuos pačius skaičiavimo pajėgumus. Pabaigoje autoriai pabrėžia, kad sprendžiant 2 klasių klasifikavimo uždavinį (yra arba nėra įtrūkimo betone) bei ieškant įtrūkimų vaizduose, jų sukurtas 1D-CNN-LSTM modelis pasiekia panašius rezultatus, kaip dažniausiai šiai problemai naudojamas 2D-CNN modelis, o straipsnio autorių sukurtas modelis pasiekia panašius rezultatus ženkliai greičiau [11].

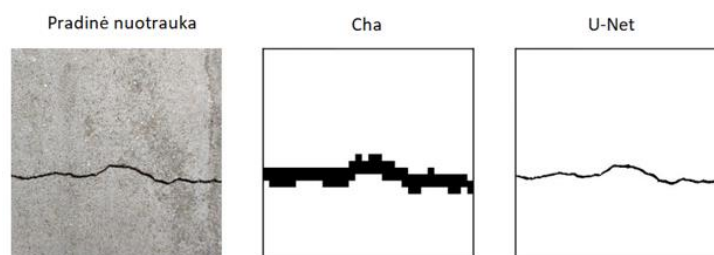


1 pav. Originali nuotrauka ir joje rasti įtrūkimai [11]

Hakaido universiteto tyrėjai, Pengyong Miao ir Teeranai Srimahachota publikuotame straipsnyje, kuriame lygina skirtingus įtrūkimų aptikimo metodus, rašo, jog dažniausiai pažeidimai betone atsiranda dėl temperatūrų svyravimų ir netinkamų priemonių naudojimui. Siekiant išvengti nelaimių, betoninės konstrukcijos reguliariai inspektuojamas. Tą galima atlikti rankiniu būdu, naudojantis vaizdų apdorojimo įrankiais, požymiais grįstu mašininu mokymusi (angl. *feature-based machine learning*) bei pasitelkiant giliojo mokymosi technologiją. Pengyong Miao ir Teeranai Srimahachota tyrimo metu visą dėmesį skyrė giliojo mokymosi modeliams. Jų tikslas buvo tarpusavyje palyginti AlexNet, GoogLeNet, Resnet18 ir VGG-16 architektūras, rasti kuri iš jų tinkamiausia įtrūkimų klasifikavimui, o kuri segmentavimui. Autoriai siekė sukurti universalų modelį, kuris įtrūkimus rastų tiek betoninėse konstrukcijose, pavyzdžiui tiltų atramosė, tiek ir betoninėse dalyse, tokiose kaip blokai. Dėl šios priežasties tyrime naudotos 69 nuotraukos padarytos laboratorijoje, kuomet skirtingų sudėčių betonas reikiamą laiką veiktas skirtingomis temperatūromis. 81 nuotrauka naudota iš betoninių konstrukcijų, tokių kaip tiltai ir pastatai. Apmokymui ir validacijai atitinkamai panaudota 58 ir 69 nuotraukos. Jos suskaidytos į mažesnius,  $256 \times 256$  formato vaizdus, kurių viso gavosi 30480. Po mažesniųjų vaizdų atskyrimo į įtrūkusius ir sveikus vykdytas tinklų apmokymas. Su AlexNet architektūra pavyko pasiekti 96,06% tikslumą, su GoogLeNet 96,69%, Resnet18 96,89%, o su VGG-16 97,4%. Apmokymas atitinkamai truko 33 val 7 min, 51 val 40 min, 60 val 22 min, 300 val 15 min. Čia straipsnio autoriai pastebi, kad VGG-16 ir pasiekė didžiausią tikslumą, visgi šio tinklo apmokymas užtruko ženkliai ilgiau negu kitų, o tikslumas taip drastiškai nesiskiria. Dėl to Pengyong Miao ir Teeranai Srimahachota nusprendė toliau nagrinėti GoogLeNet tinklą. Kitu etapu autoriai naudodami GoogLeNet tinklą, prieš tai nenaudotas nuotraukas suskaido į mažesnius  $256 \times 256$  formato vaizdus, kiekvienam iš jų skaičiuodami tikimybę, jog yra arba nėra įtrūkimo. Tokiu būdu pavykdavo teisingai suklasifikuoti 95% vaizdų, tiek darytų laboratorijoje, tiek realioje aplinkoje. Galiausiai straipsnio autoriai aprašo segmentacijos procesą. Šiai užduočiai taipogi naudojo GoogLeNet architektūrą, kaip modelio pagrindą (angl. *backbone*). Tyrimo metu sukūrus vaizdų segmentavimo įrankį, jį išbandė su likusiomis 23 apmokyte ir validacijoje nenaudotomis nuotraukomis. Kiekvienu atveju skaičiuotas Jaccard indeksas ir kiek laiko truko segmentavimas. Jaccard indekso vidurkis 80,4%, o vidutinis segmentavimo laikas 9,48 sekundės. Pabaigoje autoriai pabrėžia, jog jų iškeltoms problemoms spręsti tinkamiausia GoogLeNet architektūra, daugiausiai dėl esminio laiko skirtumo apmokant, palyginus su kitais nagrinėtais modeliais. Taipogi pažymima, kad skirtingais skaičiavimo pajėgumais rezultatai gali skirtis [12].

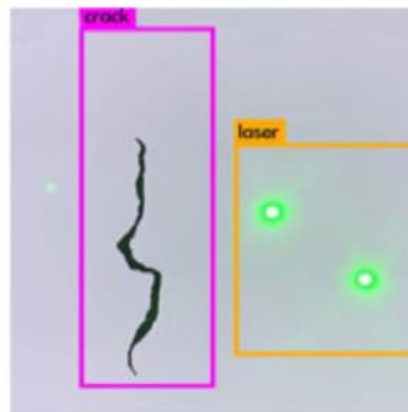
Visi pastatai per laiką artėja prie savo gyvavimo pabaigos, todėl būtina reguliariai tikrinti jų struktūros vientisumą. Tuo pačiu, senstant visuomenei didėja darbo jėgos sąnaudos, tad svarbia tyrimų krytimi tampa problema, kaip nuolat ir automatiškai stebėti pastatų konstrukcijas, pasitelkiant mažiau darbo jėgos. Anksčiau patikrų rezultatai priklausė nuo darbuotojų kompetencijos, dėl ko pagrįstai galėdavo kilti abejonių rezultatų patikimumu. Be to, atliekant tiltų arba tunelių patikrą, būtina juos uždaryti, o tai turi įtakos ne tik eismui, bet ir negarantuoja saugumo inspektoriams. Pasitaikė netgi atveju, kada vairuotojai nepastebėjo laikinų eismo apribojimų ir dėl to nukentėjo darbus vykdęs personalas. Siekiant sumažinti rizikas pastaraisiais metais vis daugiau tyrėjų iš viso pasaulio pasitelkdami skirtingas technologijas ieško geriausių sprendimų kaip pilnai arba bent jau iš dalies automatizuoti įtrūkimų betone paiešką [13, 14].

2019 metais atliktame tyrime Kinijos bei Japonijos mokslininkai Zhenqing Liu, Yiwen Cao, Yize Wang ir Wei Wang sprendžia betono įtrūkimų segmentavimo uždavinį naudodami U-Net architektūros tinklą [17]. U-Net tipo architektūra atsirado 2015 metais Freiburgo universitete, kaip metodas skirtas biomediciniui vaizdų segmentavimui, kuris vėliau pritaikytas skirtingoms segmentavimo uždavimams [15]. Toks pavadinimas pasirinktas ne atsitiktinai, nes tinklas susideda iš vaizdo sutraukiančios ir išplečiančios dalių, kas suteikia savotišką U formą [16]. Zhenqing Liu, Yiwen Cao, Yize Wang ir Wei Wang aprašytame tyrime U-Net tipo modeliui po apmokymo paduodamas  $3 \times 512 \times 512$  formato vaizdas, arba kitaip tariant spalvota  $512 \times 512$  formato nuotrauka su įtrūkimu. Modelio išvestis,  $2 \times 512 \times 512$  formato vaizdas, arba juodai balta nuotrauka, kur juodai pažymimi įtrūkimai. Neuroninis tinklas apibūdina nuokrypį tarp prognozuojamų ir faktinių modelio reikšmių, apskaičiuodamas nuostolių funkciją mokymosi metu. Kuo didesnis skirtumas tarp numatomos ir tikrosios vertės, tuo didesnė nuostolių funkcijos reikšmė, kurią reikia minimizuoti apmokymo metu. Norint pasirinkti tinkamą nuostolių funkciją, reikia atsižvelgti į gradientus, pasirinkti tinkamus mašininio mokymosi algoritmus ir duomenų tipus. Semantinio segmentavimo uždavimams dažniausiai naudojamos nuostolių funkcijos: kryžminės entropijos praradimo funkciją (angl. *cross entropy loss function*), Dice nuostolių funkciją (angl. *Dice loss function*), Lovasz funkciją ir kitas. Duomenų disbalansas yra problema, su kuria susiduriama daugelyje mašininio mokymosi uždavimų. Jis egzistuoja ir sprendžiant įtrūkimų segmentavimo uždavimą. Jei tam tikros klasės pavyzdžių yra ženkliai daugiau nei kitos, tai vadinama paprasta imtimi. Nors modelis gali gerai išspręsti paprastą uždavimą apmokymo metu, tačiau dėl didelio paprastų imčių skaičiaus indėlis į bendrą nuostolių funkciją bus labai didelis. Dėl to modelis nėra gerai apmokytas. Gali būti, kad nuostolių funkcija įstrigo prastame lokaliame optimume. Tarp geriausio vietinio optimumo ir gero vietinio optimumo gali būti didelis skirtumas. Todėl nuostolių funkcija negali priartėti prie geriausio rezultato apmokymo metu. Norint išspręsti imties disbalanso problemą, pasirenkama Focal nuostolių funkcija (angl. *Focal loss function*), kuri pagrįsta kryžminės entropijos nuostolių funkcija. Ja naudojantis tyrėjai apmokė neuroninį tinklą naudodami 84 įtrūkusio betono nuotraukas, kartu su įtrūkimų kaukėmis. Ap mokymui panaudota 57 nuotraukos, testavimui 27. Metodo palyginimui naudotas Cha aprašytas metodas, kuomet jis tą pačią uždavimą sprendžia skaidydamas nuotrauką į mažesnes dalis ir skaičiuodamas tikimybę, kad šioje vietoje yra įtrūkimas. Lyginant metodus kiekvienai nuotraukai skaičiuotas preciziškumas: teisingai teigiamų (angl. *true positive*) pikselių skaičius dalinamas iš teisingai teigiamų ir klaidingai teigiamų (angl. *false positive*) pikselių sumos. Gauta, kad testavimo imtyje vidutinis preciziškumas naudojant U-Net architektūrą 90%, o naudojant Cha metodą 82%. Tai atsispindi ir praktiniuose pavyzdžiuose (žr. 2 pav). Tad straipsnio autoriai sprendžiant semantinio segmentavimo uždavimą rekomenduoja naudoti U-Net architektūrą su Focal nuostolių funkcija [17].



2 pav. Semantinio segmentavimo rezultatai [17]

Pagal Markets and Markets prognozę, vien JAV nuo 2018 m. iki 2023 m. skirtingos civilinės infrastruktūros kasmetiniai inspektavimo kaštai išaugs nuo 1,5 milijardo dolerių iki beveik 3,4 milijardo. Didžiausią įtaką tokiam augimui turės detalesnė senesnės infrastruktūros priežiūra, mažesnę dalį sudarys augantys atlyginimai ir žaliavų kainos [18]. Siekiant sumažinti išlaidas objektų techninei priežiūrai, atliekami tyrimai norint rasti pigesnes, greitesnes ir efektyvesnes alternatyvas. Vieną tokią grįstą mašininio mokymusi ir lazerio šviesa siūlo Daejeon universiteto mokslininkai Song Ee Park, Seung-Hyun Eem ir Haemin Jeon. Jų siūloma idėja paprasta – prie drono pritvirtinama kamera, kuria fiksuojamas vaizdas ir lazeris, matuojantis atstumą iki fiksuojamo paviršiaus, bei kaip etaloną projektuoja du taškus. Kamera gauna įtrūkusio arba sveiko betono vaizdą su dviem lazerio taškais. Kitu etapu nagrinėjimą perima programinė įranga. Įvertinti, ar vaizde užfiksuotas įtrūkimas, naudojamas YOLOv3 sąsūkų neuroninis tinklas, apmokytas su 1800 nuotraukų. Tam, jog gautų didesnę tikslumą, apmokymo metu vaizdai augmentuoti sukiojant ir keičiant atspalvius. Siekiant įvertinti klasifikavimo kokybę, eksperimentiškai apskaičiuotas tikslumas 94%. Taipogi YOLOv3 modelis naudotas atvaizduojat aprėpties langą (angl. *bounding box*) apie įtrūkimą ir lazerio taškus (žr. 3 pav). Toliau vertinamas įtrūkimo ilgis ir plotis. Tam naudojami taškai gauti lazerio šviesa, kurie prieš tai sukalibruoti. Kalibravimas vyko vertinant taškų šviesos išsisklaidymą ant skirtingų paviršių projektuojant skirtingais atstumais. Žinant atstumą ir per kiek pikselių išsisklaido šviesa, galima atitinkamai įvertinti atliekant realius bandymus, kokią milimetro dalį atitinka vienas pikselis. Tokiu būdu apskaičiuojamas įtrūkimo plotis ir ilgis. Song Ee Park, Seung-Hyun Eem ir Haemin Jeon bandymus atliko su apmokymo metu nenaudotais betono vaizdais. Eksperimentuojat modelis teisingai identifikavo 92% betono įtrūkimų ir 98% lazerio taškų. Vertinant įtrūkių dydžius, vidutinė paklaida 1,5 mm. Publikacijos autoriai nurodė, kad viskas atlikta laboratorinėmis sąlygomis, o kitu tyrimų etapu numatyti bandymai su realiais visuomeninės infrastruktūros objektais [19].



3 pav. Aprėpties langai apie įtrūkimą ir lazerių taškus [19]

Tunelių statyboje betonai yra svarbiausia medžiaga. Jis ne tik formuoja tunelio skliautą, bet ir dažnai naudojamas liejant sutvirtinančiuosius poliuis [20]. Įtrūkimai – vienas iš ankstyviausių betoninių konstrukcijų degradacijos požymių, todėl įtrūkimų aptikimas ir analizė labai svarbūs siekiant užtikrinti tunelių saugumą. Rankinių plyšių patikrinimą tuneliuose dažnai atlieka apmokyti žmonės, o tai daug darbo ir laiko reikalaujantis procesas, dėl kurio darbuotojai gali rizikuoti savo gyvybe. Be to, patikrinimų rezultatai labai priklauso nuo žmogaus subjektyvumo, todėl gali atsirasti netikslumų, klaidų arba praleistų įtrūkimų [21, 22]. 2019 m. publikuota straipsnyje tyrėjai iš Kinijos Zhejiang ir Xiamen universitetų siūlo patobulintą įtrūkimų segmentavimo sprendimą, kuriam netgi suteikė CrackSegNet pavadinimą. Šis tinklas sukurtas įtrūkimų segmentavimui ir perėmė modulines kodavimo dalies projektavimo koncepcijas (angl. *modular design concepts*): ekoderio dalis, išsiplėtusios konvoliucijos (angl. *dilated convolutions*), erdvinės piramidės maksimalus telkimas (angl. *spatial pyramid max pooling*) ir praleistos jungtys (angl. *skip connections*). Pagrindinis tinklas kodavimo kelyje modifikuotas iš tipinio konvoliucinio tinklo VGG-16, kurį sudaro du  $3 \times 3$  konvoliuciniai sluoksniai ir  $2 \times 2$  maksimalaus telkimo sluoksnis. Po kiekvieno konvoliucinio sluoksnio seka ReLU aktyvinimo funkcija. Po keturių konvoliucijos ir maksimalaus telkimo sluoksnių, įvedamas savybių vektorius į keturis išplėstinius konvoliucinius sluoksnius, kurių išsiplėtimas 2, 2, 4, 4 [23]. Išplėstinis konvoliucinis sluoksnis - metodas, kuris praplečia įvestį, įterpiant tuščius laukus, tarp iš eilės einančių elementų [24]. Tada dekoderio dalis naudoja maksimalų erdvinės piramidės telkimą, kad surinktų skirtingus subregionų vaizdus. Po to seka atranka ir sujungimas, kad būtų sudarytos funkcijos, apimančios tiek lokalią, tiek globalią informaciją. Požymių žemėlapiai po kiekvienos kodavimo dalies atrenkami ir sujungiami su ypatybių žemėlapiu. Galiausiai, objektų žemėlapis įterpiamas į tris konvoliucinius sluoksnius, kad būtų gautas galutinis pikselių nustatymas. Tyrime naudotos 409 4032 × 3016 formato nuotraukos darytos skirtingose Kinijos tuneliuose. Prieš pradėdant apmokymą, paruoštos kaukės, kur kiekvienas įtrūkimo pikselis pažymėtas juodai, o sveika betono dalis baltai bei vaizdai apkarpyti iki 512 × 512 formato, tam, jog pavyktų apmokyti tinklą su turimais skaičiavimo resursais. Viso gavosi 919 vaizdų, iš kurių 736 naudoti apmokymui, likę testavimui. Tyrėjai sukurtą modelį lygino su U-Net architektūros tinklu. Lyginimui naudojo tikslumo, IoU (angl. *Intersection over Union*) pikselių tikslumo (angl. *pixel accuracy, PA*) ir atgaminimo (angl. *recall*) metrikas. Pagal visas metrikas CrackSegNet geriau atliko įtrūkimo segmentavimo užduotį (žr. 1 lentelę) [23].

**1 lentelė.** U-Net ir CrackSegNet modelių palyginimai [23]

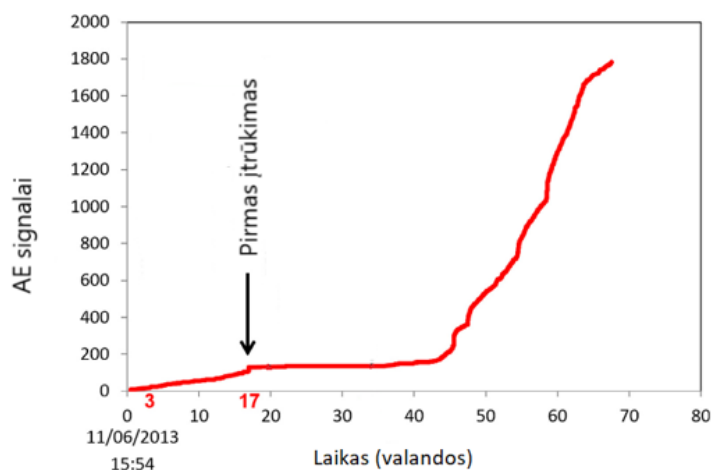
Modelis	PA	IoU	Tikslumas	Atgaminimas
U-Net	98,88%	47,64%	66,49%	60,02%
CrackSegNet	99,12%	57,28%	74,84%	70,64%

Apie 30% iš 600000 Jungtinėse Amerikos Valstijose esančių tiltų turi vienokį arba kitokį betoninį antstatą [25]. Betoniniams tiltams reguliariai atliekami patikrinimai, siekiant įvertinti bendrą būklę ir nustatyti paviršiaus įtrūkimus ar kitokį susidėvėjimą. Tokio tipo žmonių atliekami patikrinimai brangūs, reikalaujantys daug laiko ir darbo jėgos. Autonominiai patikrinimai potencialiai perspektyvi žmogiškų patikrinimų alternatyva [26 – 28]. Sattar Dorafshan, Robert J. Thomas, Marc Maguire 2018 metais publikuotame tyrime lygina kraštų aptikimo (angl. *edge detector*) metodų ir mašininio mokymosi algoritmų pritaikomumą ieškant įtrūkimų betone. Tyrėjai pasirinkimą įtrūkimų betone paieškai grindžia tuo, jog dažnai tiltų konstrukcijos fotografuojamos didelės raiškos kameromis, o vėliau kiti darbuotojai rankiniu būdu nagrinėja šiuos vaizdus. Siekiant palengvinti ir pagerinti analizę, naudojami skirtingi filtrai, kurie paryškina įtrūkimus. Iš dalies tai galima pavadinti savotiška kampų detekcija, nes pavyzdžiui nuotrauką pavertus juodai balta, įtrūkimas dažniausiai suprantamas kaip kraštas. Straipsnio autoriai kraštų aptikimui naudojo Roberts, Prewitt, Sobel, LoG, Butterworth ir Gauso filtrus. Tyrėjai mašininio mokymosi nagrinėjimui pasirinko AlexNet tinklą, kurį apmokė su 319 įtrūkusio betono vaizdų ir su 3101 sveiko betono vaizdais. Taipogi AlexNet tinklas apmokytas skirtingais būdais. Pirmuoju, kai pradiniai svoriai priskiriami atsitiktinai, o vėliau mokymosi metu jie koreguojami. Kitu atveju, kada svoriai priskiriami pagal jau apmokytą tinklą kitiems vaizdams, kaip pavyzdžiui ImageNet duomenų rinkiniui, tačiau apmokymu metu keisti svoriai pagal turimus vaizdus. Toks metodas vadinamas perkeltu mokymusi (angl. *transfer learning*). Kraštų aptikimo metodai lyginti tarpusavyje skaičiuojant tikslumo metrikas pikseliams, t.y. teisingai klasifikuotų pikselių suma dalinama ir visų vaizde esančių pikselių skaičiaus. Skirtingais filtrais gauti tikslumai: Roberts – 95%, Prewitt – 97%, Sobel – 97%, LoG – 98%, Butterworth – 96% ir Gauso 95%. Tyrėjai komentuodami šiuos rezultatus pabrėžia, jog reikia atsižvelgti į tai, kad buvo didžiulis netolygumas, kai daug didesnę dalį nuotraukų sudarė vaizdai be įtrūkimų. AlexNet tinklo rezultatai vertinti skaičiuojant teisingai suklasifikuotų, iki tol nenaudotų vaizdų, dalį. Apmokant modelį atsitiktinai priskyrus svorius, gautas 97,5% tikslumas, per 1 val 43 min apmokymo. Naudojant perkeltą apmokymą 98,1%, apmokius per 1 val 8 min ir su tiek pat epochų. Sattar Dorafshan, Robert J. Thomas ir Marc Maguire publikacijos pabaigoje siūlo naudoti tiek skirtingus filtrus, siekiant rasti įtrūkimus pikselių lygyje, tiek mašininio mokymosi metodus, norint klasifikuoti sveiką ir įtrūkusį betoną [29].

Rečiau sutinkama, bet labai aktuali, betono įtrūkimų problema iškyla kalbant apie panaudoto branduolinio kuro saugojimą. Prieš palaidojant panaudoto kuro kapsules giliai pro žemę, jos patalpinamos į superkonteinerį [30, 31]. Dažniausiai jis sudarytas iš keleto sluoksnių: kuro pakuotės, pakuočių užpildo, betoninio buferio ir priklausomai nuo to kur bus laidojimas dar gali turėti ir metalinį apvalkalą. Betoninio buferis sukuria šarminę aplinką, užtikrinančią lėtą ir vienodą išorinės pakuotės koroziją bei radiacinį ekranavimą [32]. Visgi per laiką betonas dėvisi ir jame atsiranda įtrūkimų. Pro juos į aplinką gali veržtis radioaktyvios dalelės. Siekiant to išvengti, Briuselio universiteto mokslininkai S. Iliopoulos, D.G. Aggelis, L. Pyl, J. Vantomme, P. Van Marcke, E. Coppens ir L. Areia siūlo sprendimą, kaip galima stebėti superkonteinerių būklę. Tyrėjai išmėgina tris metodus: skaitmeninio vaizdo koreliaciją (angl. *Digital Image Correlation, DIC*), akustinę emisiją (angl. *Acoustic Emission, AE*) ir ultragarso impulso greitį (angl. *Ultrasonic Pulse Velocity, UPV*) [33]. Skaitmeninio vaizdo koreliacija, tai optinis, nekontaktinis, neardomasis metodas, leidžiantis identifikuoti poslinkius tarp tolesnių fotoaparatu padarytų vaizdų. Naudojant kamerą užfiksuojami keli tiriamo objekto vaizdai. DIC įterptoji programinė įranga identifikuoja kiekvieną vėlesnių vaizdų tašką, tirdama jo kaimynystę. Po vaizdų apdorojimo galiausiai sukuriamas poslinkio ir deformacijos laukas, taip pat vizualizuojami defektai, pateikiant vietą, ilgį, plotį ir kryptį [34, 35].



Akustinė emisija - realaus laiko neardomoji technika paremta tuom, jog bet koks lūžis išskiria energiją, kuri sklinda tamprių bangų pavidalu visomis kryptimis. Kai banga pasiekia paviršių, atitinkami jutikliai registruoja trumpalaikį atsaką [36]. Ultragarso impulso greičio metodas pagrįstas terpėje sklindančios bangos greičio matavimu, kuris yra aiškiai koreliuoja su medžiagos tankiu ir tamprumo savybėmis [37]. Straipsnio autoriai bandymus skirtingais metodais pradėjo iš karto po to, kai panaudotas branduolinis kuras buvo užsandarintas superkonteineryje. DIC metodui betoninė dalis fiksuota vieną kartą per dieną. Tokiu būdu pastebėta, kad pirmasis įtrūkimas atsirado pirmą parą. Siekiant nustatyti tikslų laiką, kada įtrūko betonas, išanalizuoti AE metodo duomenys (žr. 4 pav.) Gauta, jog pirmas įtrūkimas betoninėje superkonteinerio dalyje atsirado po 17 valandų. Paskutiniu etapu norėta nustatyti įtrūkimų ilgį ir plotį. Tam pasitelktas UPV metodas. Pritvirtinus ultragarso siųstuvą ir imtuvą maždaug toje vietoje kur atsirado įtrūkimas (vieta nustatyta vizualiai), atlikti ultragarsiniai matavimai. Gauta, jog įtrūkimo ilgis 38 cm, plotis 15  $\mu\text{m}$ . Siekiant patikrinti šiuos matavimus, superkonteineris padengtas ultravioletinėje šviesoje spindinčiais dažais ir apšviestas UV spinduliais. Matavimai gauti su UV spinduliais patvirtino duomenis gautus UPV metodu. Straipsnio pabaigoje autoriai pažymi, jog visi bandymai atlikti laboratorinėmis sąlygomis ir reikėtų papildomų eksperimentų prieš pradėdant realiai naudoti šiuos superkonteinerių stebėjimų metodus [33].



4 pav. AE signalų rezultatai [33]

Besivystančiose valstybėse, ženkli dalis civilinės infrastruktūros biudžeto tenka esamų objektų priežiūrai. Dėl šios priežasties vis ieškoma būdų kaip sumažinti išlaidas taip, jog priežiūros kokybė nenukentėtų [38]. Dažnai naudojami skaitiniai metodai, kuriais modeliuojami galimi įtrūkimai betone ir vėliau visa tai išbandoma laboratorijoje su skirtingomis betono rūšimis [39]. Taipogi ne retai atliekami ultragarsiniai tyrimai, kada betoninėmis konstrukcijomis leidžiami signalai ir vertinant laiką tarp signalo išsiuntimo ir priėmimo nustatomi konstrukcijos pažeidimai [40]. Atsiradus giliajam mokymuisi (angl. *deep learning*) prasidėjo tyrimai, o vėliau ir taikymai, kaip greičiau ir pigiau aptikti betoniniu konstrukcijų pažeidimus. Siūlymą, kaip panaudojant skirtingas betono nuotraukas apmokyti, o vėliau aptikti konstrukcijos pažeidimus siūlo Jose A. Guzman-Torres, M.Z. Naser ir Francisco J. Domínguez-Mota 2022 m. publikuotame straipsnyje. Pradžioje tyrėjai surinko 780 256

× 256 formato benoto nuotraukų, iš kurių 300 su įtrūkimu ir 300 be, naudota apmokymui. Likę 180 vaizdų skirti modelių testavimui, iš kurių pusė turėjo įtrūkimus, pusė ne. Nuotraukose betono įtrūkimai yra nuo 0,06 mm iki 25 mm pločio. Taip galima aptikti smulkesnius įtrūkimus, kas leidžia anksčiau pastebėti pažeistą konstrukciją. Tam, kad galėtų įvertinti kuris modelis tinkamiausias įtrūkimų paieškai, straipsnio autoriai vertino tikslumą, gautą po modelio išbandymo su testavimo imtimi bei apmokymo laiką su turimais skaičiavimo resursais. Rezultatai pateikiami 2 lentelėje. Geriausias rezultatas pasiektas su VGG-16 modeliu. Autoriai pažymi, kad jų atliktas tyrimas nuo panašaus tipo bandymų skiriasi tuo, jog apmokymui naudoti ir ypač maži betono įtrūkimai, kurie nesiekia net 0,5 mm pločio. Pasak jų, tai suteikia galimybę pokyčius betone pastebėti ankstyviausioje stadijoje [41].

**2 lentelė.** Skirtingų modelių detalės ir tikslumas [41]

<b>Modelis</b>	<b>Viso parametrų</b>	<b>Apmokomi parametrai</b>	<b>Neapmokomi parametrai</b>	<b>Modelio dydis, Mb</b>	<b>Tikslumas,%</b>	<b>Apmokymo laikas, s</b>
VGG-16	17926209	3211521	14714688	57,1	97,78	231,9
MobileNet	9651649	6422785	3228864	16,82	94,44	105,4
ResNet50	36433025	12845313	23587712	92,43	88,89	197,6
DenseNet121	13460289	6422785	7037504	29,3	58,88	230,4
InceptionV3	28356641	6553857	21802784	85,85	49,44	107,9

## 2. Duomenys ir tyrimo metodai

### 2.1. Duomenys

Baigiamajame magistro projekte naudoti viešai prieinami vaizdai. Nuotraukos modelių apmokymui naudotos iš viešai pasiekiamo duomenų rinkinio „Surface Crack Detection“. Šiame rinkinyje pateikta 40000 skirtingo tipo, cheminių savybių ir apšvietimo betono vaizdų. 20000 256 × 256 formato nuotraukos yra su įtrūkimu, o 20000 tokio paties formato nuotraukų yra be įtrūkimo (žr. 5 pav.). Modelių testavimui naudotos aukštos raiškos, 1024 × 1024 formato nuotraukos išskirtinai bent su vienu įtrūkimu betone (žr. 6 pav.). Šios nuotraukos yra iš visiškai kitos imties.

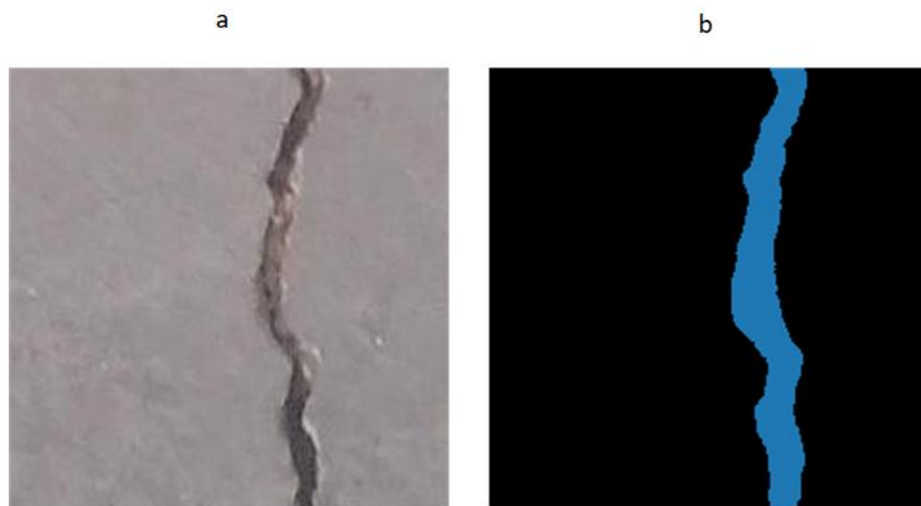
Prieš apmokant modelius semantiniam įtrūkimų segmentavimui, 100-tui atsitiktinai atrinktų įtrūkusio betono vaizdų, iš turimų 20000, naudojant hasty.ai įrankį, sukurtos kaukės (žr. 7 pav.).



5 pav. Įtrūkusio a ir sveiko b betono pavyzdys iš apmokymo imties



6 pav. Didelės raiškos betono su įtrūkimu pavyzdys



7 pav. Įtrūkimo a ir jo kaukės b pavyzdys

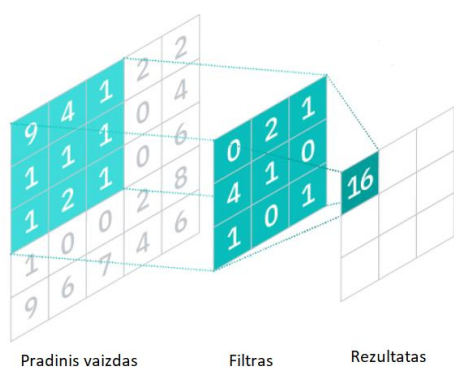
## 2.2. Tyrimo metodai

Tyrimo metu vaizdų analizei naudoti sąsūkų neuroniniai tinklai (angl. *convolutional neural network*, *CNN*). Pasitelkiant CNN spręstos klasifikavimo ir semantinio segmentavimo užduotys. Klasifikavimui naudotos EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB, AlexNet ir VGG-16 architektūros. Semantiniam segmentavimui naudota U-Net architektūra.

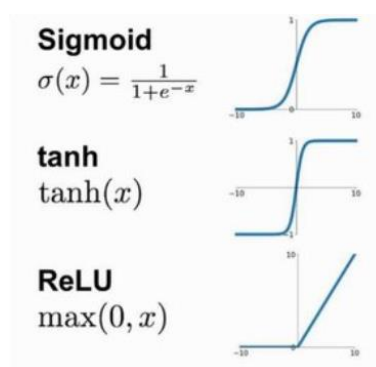
## 2.2.1. CNN

Sąsūkų neuroninis tinklas – dirbtinio neuroninio tinklo porūšis, dažniausiai naudojamas vaizdų analizei. Šis metodas įkvėptas žmogaus regėjimo, kai stebint aplinką suvokiama visuma pagal atskiras sritis, todėl paprastai dėmesys atkreipiamas į konkrečias dalis pagal poreikius, o vėliau atskiros dalys sisteminamos į bendrą visumą [42, 43].

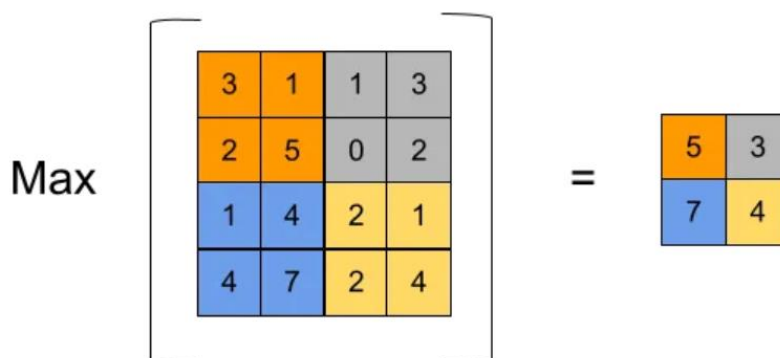
Veikimas paremtas tuom, jog vaizdai paveikiami skirtingais filtrais, iš kurių kiekvienas ieško jam būdingų požymių, pavyzdžiui linijų, kampų, atspalvių ir pan. Sąsūkos metu, informacija iš dvejų skirtingų šaltinių sutraukiama į vieną. Šiuo atveju tai reiškia, jog vaizdo pikselių reikšmės, dauginamos iš filtro reikšmių (žr. 8 pav.). Kadangi įprastai filtras būna mažesnis negu vaizdas, tad tuo pačiu informacija suspaudžiama. Siekiant vienu sričių svarbą sustiprinti, o kitų sumažinti, naudojamos aktyvacijos funkcijos. Dažniausiai naudojamos Sigmoid, tanh ir ReLU (žr. 9 pav.). Po sąsūkos ir aktyvacijos operacijų atliekama telkimo operacija (angl. *pooling*). Priklausomai nuo telkimo tipo, maksimalaus, minimalaus, vidutinio (angl. *max pooling*, *min pooling*, *average pooling*) ir nuo telkimo ploto, išsaugoma tik reikiama informacija. 10 paveiksle pateiktas maksimalaus telkimo pavyzdys su  $2 \times 2$  dydžio filtru ir žingsniu (angl. *stride*) per 2 pikselius. Sąsūkų, aktyvacijos ir telkimo kombinacija kartojama daug kartų su skirtingais to pačio daikto vaizdais, kol išskiriamos svarbiausios objekto savybės. Tada turima informacija išlyginama (angl. *flatten*) ir perduodama dirbtiniam neuroniniam tinklui apsimokyti (žr. 11 pav.) [43].



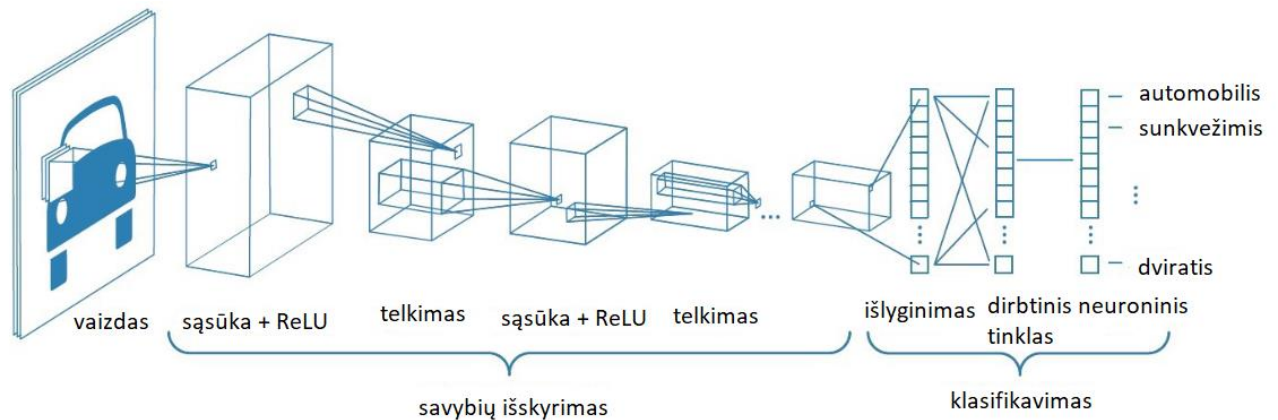
8 pav. Sąsūkos operacija [44]



9 pav. Aktyvacijos funkcijos [45]



10 pav. Maksimalaus telkimo pavyzdys [46]



11 pav. Sąsūkų neuroninis tinklas [47]

### 2.2.1.1. Stochastinis gradientinis nusileidimas

Stochastinis gradiento nusileidimas (SGD) yra mašininio mokymosi algoritmas, kuris naudojamas kaip optimizatorius, norint keisti svorio  $W_i$  ir poslinkio (angl. *bias*)  $b$  reikšmes kiekvienoje iteracijoje apmokymo metu. Tai padeda sumažinti paklaidą tarp numatomos ir tikslios reikšmės. Šio algoritmo tikslas – optimizuoti neuroninį tinklą, atgalinio sklidimo (angl. *back-propagation*) metu. Be to, algoritmas atgalinio sklidimo metu dažnai tiria parametrų gradientus, susijusius su svoriu  $W$ , jog gautų minimalius nuostolius. Tai sukuria stabilų paklaidų gradientą ir stabilią konvergenciją. Be to, tai bus naudojama siekiant didesnio išvesties tikslumo, atnaujinant svorio reikšmes paslėptuose sluoksniuose. Svoriai apskaičiuojami:

$$W_i = W_i - L_r \frac{\delta paklaidos}{\delta W_i} L(W_0, W_1, \dots, W_n) \quad (2.1)$$

Kur  $L$  yra nuostolių funkcija ir kiekviena svorio reikšmė ( $W_0, W_1, \dots, W_n$ ) iš fiksuoto mokymosi žingsnio (angl. *learning rate*), jog gautų naujas svorių reikšmes. Svorijų ir poslinkio reikšmės apskaičiuojamos naudojantis grandinės taisykle (angl. *chain rule*):

$$\frac{\delta paklaidos}{\delta W_i} = I_i * \frac{\delta paklaidos}{\delta O_{tikra}} \quad (2.2)$$

$$\frac{\delta paklaidos}{\delta b} = \frac{\delta paklaidos}{\delta O_{tikra}} \quad (2.3)$$

Kur  $I_i$  – įvesčių skaičius,  $O_{tikra}$  – tikroji išvesties reikšmė [48].

### 2.2.1.2. ADAM optimizatorius

Adaptyvus impulso įvertinimas (angl. *Adaptive momentum estimation, Adam*) – tai pažangus modelio parametrų optimizavimo algoritmas, apjungiantis AdaGrad ir RMSPro algoritmų pranašumus. Palyginus su stochastinio gradiento nusileidimo algoritmu, Adam nelinkęs užstrigti lokaliame optimume ir jo atnaujinimo greitis didesnis. Be to, Adam parametrai gana paprasti, todėl pakanka suderinti keletą hiperparametrų. Darant prielaidą, kad  $\theta_t$  yra optimizuojamas parametras  $t$  laiko žingsnyje. Kol  $\theta_t$  nekonverguoja, Adam algoritmo atnaujinimo taisyklės galima išreikšti taip:

$$g_t = \nabla_{\theta} f_t(\theta_{t-1}) \quad (2.4)$$

$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t \quad (2.5)$$

$$v_t = \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2 \quad (2.6)$$

$$m'_t = \frac{m_t}{1 - \beta_1^t} \quad (2.7)$$

$$v'_t = \frac{v_t}{1 - \beta_2^t} \quad (2.8)$$

$$\theta_t = \theta_{t-1} - \frac{\alpha}{\sqrt{v'_t} + \epsilon} \cdot m'_t \quad (2.9)$$

kur  $m_t$  yra poslinkio pirmosios akimirkos įvertis (angl. *biased first moment estimate*);  $v_t$  poslinkio antrojo neapdorotas momento įvertis (angl. *biased second raw moment estimate*);  $\beta_1$  ir  $\beta_2$  pirmojo ir antrojo momento įverčių eksponentinis mažėjimo greitis (angl. *exponential decay rates*);  $m'_t$  yra poslinkio koreguotas pirmosios akimirkos įvertis (angl. *bias-corrected first moment estimate*);  $v'_t$  poslinkio pataisytas antrasis neapdorotas momento įvertis (angl. *bias-corrected second raw moment estimate*);  $f_t(\cdot)$  stochastinė tikslo funkcija;  $g_t$  – gradientas stochastinio tikslo atžvilgiu;  $\alpha$  yra pradinis mokymosi žingsnis;  $\epsilon$  yra labai mažas teigiamas skaičius, kuris naudojamas tam, kad vardiklis nebūtų lygus nuliui [49].

### 2.2.2. Architektūros

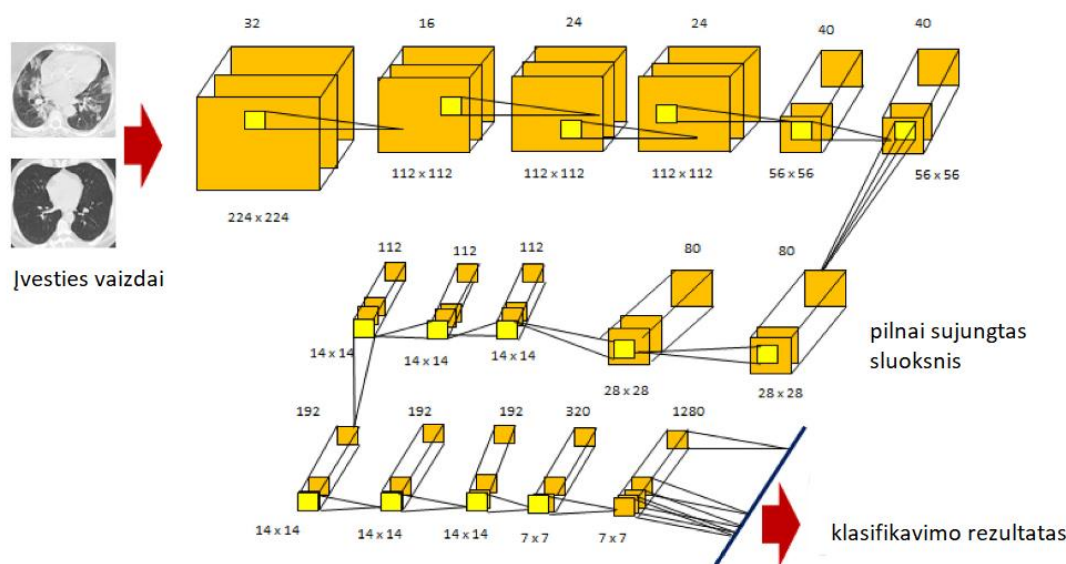
Modelio architektūra – visuma sluoksnių ir veiksmų, kurie leidžia apdoroti apmokymo duomenis taip, kad vėliau sistema savarankiškai galėtų priimti sprendimus. Tam tikros architektūros modeliai turi savo pavadinimus. Jie dažniausiai suteikiami dideliems modeliams, kurie gerai sprendžia mašininio mokymosi užduotis ir plačiai naudojami skirtingoms užduotims [50].

#### 2.2.2.1. EfficientNet

EfficientNet architektūra siūlo nestandartinį neuroninio tinklo modelių mastelio keitimo metodą, kuriuo didinamas gylis, plotis ir tikslumas. EfficientNet mastelio keitimo metodas nuosekliai didina tinklo plotį, gylį ir skiriamąją gebą, naudojant iš anksto nustatytų mastelio parametrų rinkinį. Šio modelio autoriai pradėjo nuo bazinio tinklo vystymo, naudodami neuroninės architektūros paiešką (angl. *neural architecture search*), kuri automatizuoja neuroninio tinklo kūrimą. Naudojant slankiojo kablelio operacijas per sekundę (angl. *floating-point operations per second*), jos optimizuoja metodo efektyvumą ir tikslumą. Šioje architektūroje naudojama kilnojamos inversijos kliūtis konvoliucija



(angl. *the movable inversion bottleneck convolution, MBConv*). 12 paveiklėse pavaizduota tiksliausiai klasifikavimo užduotis atliekanti EfficientNetB0 architektūra. Joje yra 18 sąsūkos sluoksnių, kurių kiekvienas turi  $3 \times 3$  arba  $5 \times 5$  branduolį. Įvesties vaizdo dydis –  $224 \times 224$  pikseliai. Kituose sluoksniuose skiriamoji geba sutraukiama, kad sumažėtų objektų žemėlapių (angl. *feature map*) dydis, bet padidintas plotis, siekiant pagerinti tikslumą. Parametrai nustatomi modelio apmokymo metu. Siekiant išvengti modelio persimokymo, naudojamas išankstinis sustabdymas (angl. *early stopping*) [51].



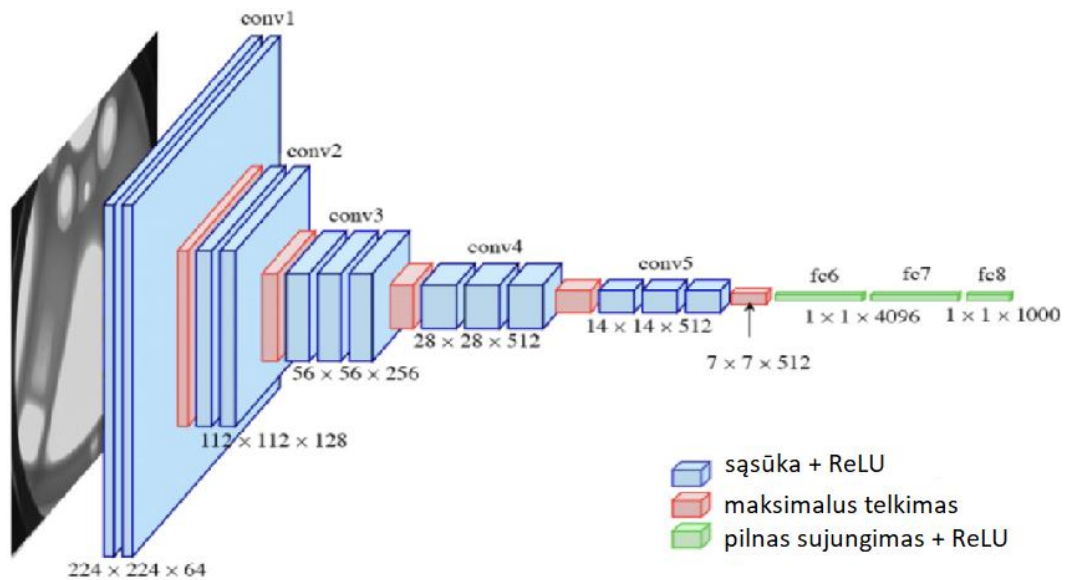
12 pav. EfficientNet architektūros schema [51]

### 2.2.2.2. VGG-16

Vizualios geometrijos grupė (angl. *visual geometry group, VGG*) yra tam tikras sąsūkų neuroninis tinklas kurį pasiūlė K. Simonyan ir A. Zisserman iš Oksfordo universiteto [52]. VGG-16 vadinamas modelis, sudarytas iš 16 sluoksnių. Dėl to šis modelis tampa dideliu, turinčiu apie 138 milijonus apmokomų parametrų [53]. VGG vidutiniškai pasiekia 92,7% tikslumą imant 5 geriausius klasifikacijos testus ImageNet duomenų rinkinyje, kuriame yra daugiau nei 14 milijonų vaizdų iš 1000 klasių. Laipsniškas semantinio kompleksumo (angl. *semantic complexity*) didėjimas, didėjant tinklo gyliui, yra pagrindinė VGG tipo sąsūkų neuroninio tinklo savybė. Šis hierarchinis ypatumas palengvina išskirtų savybių (angl. *extracted features*) pritaikymą įvairiuose duomenų rinkiniuose ir užduotyse [54]. Įrodyta, kad VGG tipo architektūros tinklai gerai klasifikuoja ne tik vaizdus iš „ImageNet“ duomenų rinkinio [55].

VGG-16 įvestis yra fiksuoto dydžio  $224 \times 224$  RGB vaizdas. Vaizdas perduodamas per mažėjančio dydžio sąsūkų sluoksnius, naudojant  $3 \times 3$  filtras. Erdvinis telkimas (angl. *spatial pooling*) atliekamas su penkiais maksimalaus telkimo sluoksniais, kurie eina po kai kurių sąsūkos sluoksnių, taip sumažinant informacijos dydį. Po sąsūkos sluoksnių naudojami trys pilnai sujungti sluoksniai: pirmieji du turi po 4096 kanalus, trečiasis atlieka 1000 galimų variantų klasifikaciją, todėl jame yra 1000 kanalų. Paskutiniame sluoksnyje naudojant softmax funkciją atliekamas klasifikavimas (žr. 13 pav.)[54].



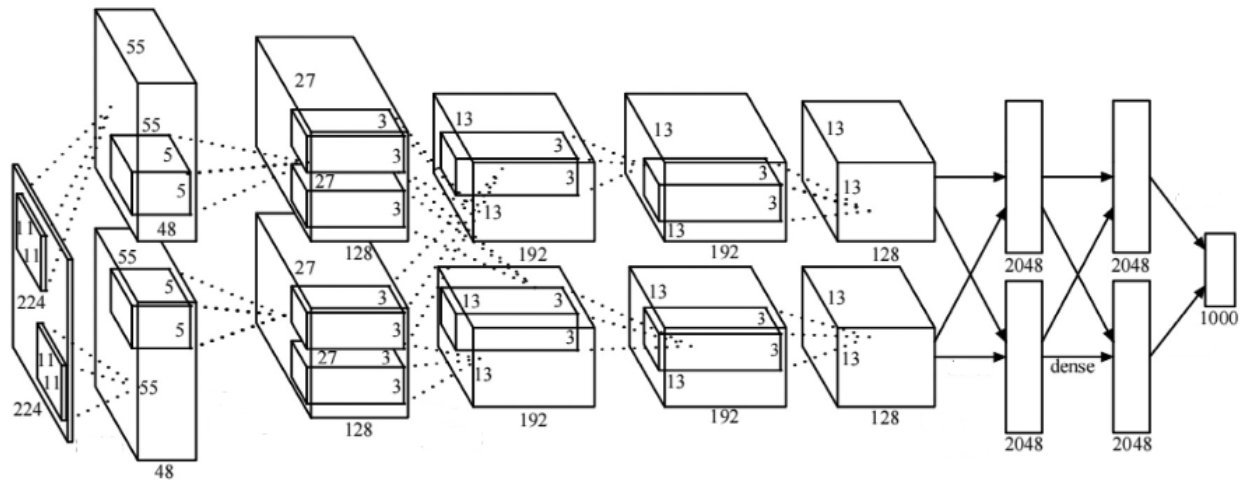


13 pav. VGG-16 tinklo architektūra [56]

### 2.2.2.3. AlexNet

AlexNet architektūra pirmą kartą panaudota dar 2012 m. Alekso Krizhevskio, dėl ko modelis ir pavadintas kaip „Aleksio tinklas“ [57]. AlexNet remiasi tiesioginiu sklidimo tinklu (angl. *forward propagation network*), ir apima 10 sluoksnių, tokių kaip įvesties, sąsūkos, atsako normalizavimo (angl. *response-normalization*), maksimalaus telkimo bei išvesties sluoksniai. AlexNet viso turi 8 sąsūkos sluoksnius. Pirmieji 5 yra sąsūkos sluoksniai, tarp kurių 3 maksimalaus telkimo, o paskutiniai 3 pilnai sujungti sluoksniai. Modelis turi apie 60 mln. apmokomų parametrų [58].

AlexNet modelio įvesties sluoksnis yra spalvotas  $224 \times 224$  dydžio vaizdas. Tiek viršutinėje, tiek apatinėje pirmojoje sąsūkos sluoksnio dalyje, yra po 48 sąsūkos branduolius, kurių formatas  $11 \times 11$ , su 4 pikselių žingsniu. Tada antriniame tinkle gaunami 48  $55 \times 55$  formato požymių žemėlapiai. Vėliau naudojama ReLU aktyvacijos funkcija, vietinio atsako normalizavimo (angl. *local response-normalization, LRN*) sluoksniai ir maksimalaus telkimo sluoksnis, kurio dydis  $3 \times 3$ , su 2 pikselių žingsniu. Tada kiekviename antriniame tinkle gaunami po 48,  $27 \times 27$  dydžio požymių žemėlapiai, kurie laikomi antrosios sąsūkos įvestimi. Šis sluoksnis, kurio dviejuose antriniuose tinkluose yra 128  $5 \times 5$  su 1 pikselio žingsniu, branduoliai, naudodami tą patį užpildą (angl. *padding*), išveda 128 požymių žemėlapius. Po ReLU, LRN ir maksimalaus telkimo operacijų, sudaromi 128 požymių žemėlapiai, kurių dydis  $13 \times 13$ . Trečias, ketvirtas ir penktas sąsūkos sluoksniai sujungti vienas su kitu be jokių įsiterpusio telkimo (angl. *intervening pooling*) ar normalizavimo sluoksnių. Trečiame sąsūkos sluoksnyje naudojami 384  $3 \times 3 \times 256$  dydžio branduoliai, sujungti su antrojo sąsūkos sluoksnio išėjimu. Ketvirtasis sąsūkos sluoksnis turi 384 branduolius, kurių dydis yra  $3 \times 3 \times 192$ , o penktasis sąsūkos sluoksnis turi 256 branduolius, kurių dydis  $3 \times 3 \times 192$ . Septintasis ir aštuntasis sluoksniai pilnai sujungti ir turi po 4096 neuronus. Paskutinis sluoksnis yra išvestis su reikiamu neuronų skaičiumi (žr. 14 pav). Magistro baigiamajame projekte AlexNet paskutiniame sluoksnyje turėjo 2 neuronus, kadangi nagrinėta 2 klasių klasifikavimo problema [58].

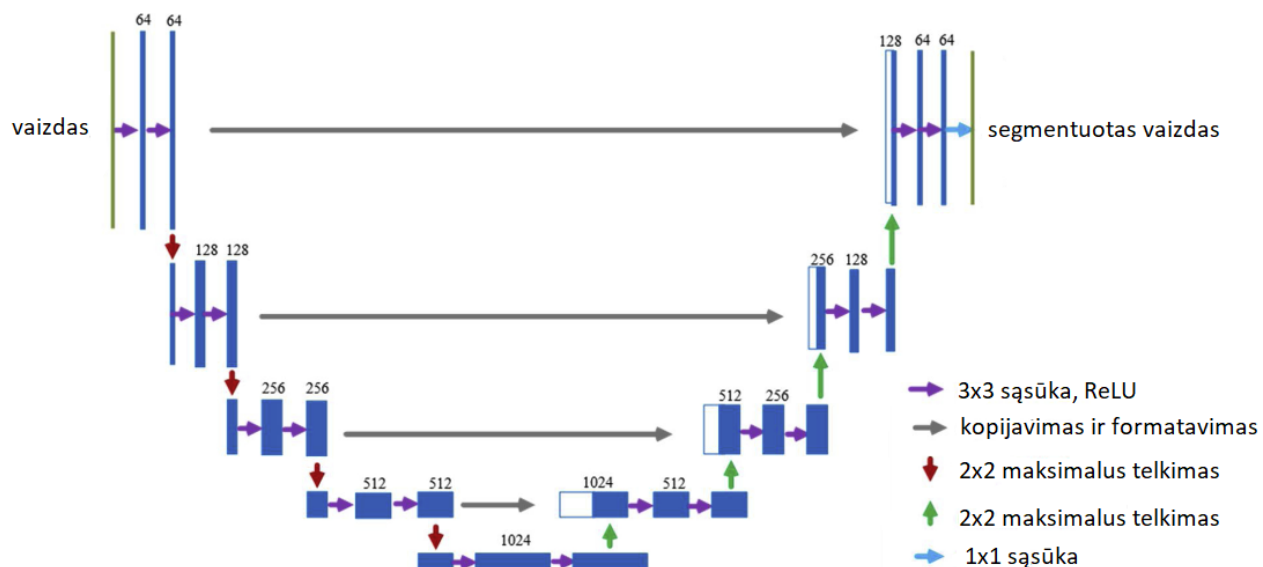


14 pav. AlexNet architektūros schema [58]

#### 2.2.2.4. U-Net

U-Net yra visiškai konvoliucinis tinklas (angl. *fully convolutional network*, FCN), kurį pirmasis aprašė O. Ronneberger ir kt. [15]. U-Net architektūrą sudaro dvi dalys: sutraukimas ir išplėtimas. Sutraukimo dalis, arba enkoderis (angl. *encoder*), naudojama užfiksuoti kontekstą paveikslėlyje ir tuo pačiu atsakinga už automatinę savybių gavybą. Išplėtimo dalis, vadinama dekoderiu (angl. *decoder*), naudojama tiksliam lokalizavimui, pasitelkiant transponuotą sąsūką. Siekiant lokalizuoti ir atrinkti ypatybes, išsiplėtimo dalis sujungta su didelės raiškos savybėmis iš sutraukimo dalies per praleistas jungtis (angl. *via skip-connections*). Modelio išvestis – kaukė, kuri rodo numatomą kiekvieno pikselio klasę. Taipogi U-Net architektūra patikimai atlieka segmentavimo užduotis esant ribotam kiekiui duomenų [59].

Kiekvienas enkoderio blokas susideda iš dviejų nuosekliai normalizuotų,  $3 \times 3$  sąsūkos sluoksnių. Po jo seka maksimalus telkimo sluoksnis, su  $2 \times 2$  dydžio, 2 pikselių žingsnio branduoliu, naudojamas įvesties vaizdo pavyzdžiui sumažinti ir gauti daugiau kontekstinės ir erdvinės informacijos. Išplėtimo dalyje naudojami tie patys enkoderio blokai, išskyrus tai, jog panašios rezoliucijos savybių žemėlapiai, iš žemesnės atrankos dalies (angl. *down-sampling*), sujungiami su savybių žemėlapiams iš aukštesnės atrankos dalies (angl. *up-sampling*) po to, kai jie apkarpmi ir suderinami, taip, jog sutaptų matmenys. Praleistos jungtys naudojamos norint gauti tikslesnę informaciją, kuri galėjo būti prarasta tarpiniuose etapuose. Po kiekvieno dekoderio bloko, savybių žemėlapiams atliekamas išdidinimas, siekiant padidinti skiriamąją gebą (žr. 15 pav.). Dekoderis semantiškai projektuoja enkoderio išmuktas, mažesnės skiriamosios gebos ypatybes į didesnės skiriamosios gebos pikselių erdvę, jog tankiai klasifikuotų taškus. Visgi, labai sumažėję savybių žemėlapiai dėl naudojamos ne visos informacijos, dažnai praranda dalį erdvinės skiriamosios gebos, dėl kurios atsiranda vaizdo šiurkštumas ir gauna mažiau informacijos apie objekto kraštą [59].



15 pav. U-Net architektūra [59]

### 2.2.3. Tyrimo eiga

Magistro baigiamasis projektas sudarytas iš 2 dalių: įtrūkimų klasifikavimo ir jų segmentavimo. Pirmojoje dalyje išbandyti EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB3 modeliai apmokymo metu tiek naudojant, tiek nenaudojant vaizdų augmentaciją juos sukant 90°. Taipogi ieškant geriausio klasifikatoriaus išbandyti ir AlexNet bei VGG-16 modeliai, be vaizdų augmentacijos. Siekiant gauti stabilius tikslumo įverčius, naudojama 5 dalių kryžminė validacija, o darbe pateikiamas visų 5 dalių testavimo tikslumo metrikų vidurkis.

Įtrūkimų segmentavimui naudoti du metodai. Pirmasis, kada vaizdas su įtrūkimu skaidomas mažesnėmis dalimis ir kiekvienai iš šių dalių apskaičiuota tikimybė, jog joje yra įtrūkimas. Vertinant tikimybę, kad mažesnėje vaizdo dalyje yra įtrūkimas, naudotas geriausias klasifikavimo modelis ir ansamblis, sudarytas iš 3 geriausių klasifikatorių. Modelių ansamblis, tai procesas, kurio metu apjungiami keli skirtingi modeliai, siekiant numatyti rezultatą. Ansamblis apibendrina kiekvieno bazinio modelio prognozę ir pateikia vieną, galutinį rezultatą. Jis naudojamas norint sumažinti prognozės paklaidą [60]. Vėliau atvaizduojamas šilumos žemėlapis (angl. *heatmap*). Antru metodu išbandytos skirtingos U-Net tinklo modifikacijos. Atlikti bandymai U-Net architektūra su EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB3, EfficientNetB4 ir EfficientNetB5 pagrindiniais modeliais.

### 2.2.3.1. Metodų palyginimas

Klasifikavimo modeliai tarpusavyje lyginti vertinant tikslumą testavimo imtyje, apmokymo laiką ir teisingai klasifikuotų nuotraukų dalį iš aukštos rezoliucijos vaizdų imties. Atliekant šį vertinimą, kiekvieną kartą po modelio apmokymo su juo bandyta klasifikuoti 1000, iki tol apmokyme, validacijoje ir testavime nenaudotų vaizdų. Su kiekvienu modeliu išbandytos tos pačios nuotraukos.

Modelio tikslumas apskaičiuojamas dalinant teisingai klaidingai (angl. *true negative*) ir tikrai teisingai (angl. *true positive*) klasifikuotų vaizdų skaičių, iš viso klasifikavime bandytų vaizdų skaičiaus.

Modeliai, kuriais atskirose vaizdo dalyse skaičiuojama tikimybė, jog yra įtrūkimas, tarpusavyje lyginti vizualiai.

Ieškant tinkamiausio pagrindinio U-Net architektūros modelio, vertintas apmokymo laikas, nuostolių funkcijos bei IoU reikšmės. IoU, arba aibių sankirtos ir sąjungos santykio vertė, naudojama vertinant semantinio segmentavimo modelius. Metrika rodo, kokią dalį pikselių testavimo imtyje modelis teisingai segmentavo, laikant, jog pateiktoje kaukėje pažymėti visi objekto pikseliai.

### 2.2.3.2. U-Net apmokymas

Apmokant bet kokį neuroninį tinklą, siekiama minimizuoti nuostolių funkciją. Intuityviai nuostolių funkcija gali būti laikoma klaidomis, kurias modelis daro apmokymo metu. Tokiu būdu modelis supranta, kiek prognozė nutolusi nuo pageidaujamo rezultato ir leidžia atitinkamai atnaujinti svorius. Semantinio segmentavimo užduotims dažnai naudojama kryžminės entropijos (angl. *cross entropy*) nuostolių funkcija. Kryžminė entropija, tai informacijos teorijos matas, kuriuo siekiama rasti skirtumą tarp dviejų skirstinių. Ją galima suprasti įvertinus pagrindinės tikimybės pasiskirstymą P ir pasiskirstymo Q apytikslę reikšmę. Tada Q kryžminė entropija iš P yra papildomų bitų skaičius, vaizduojantis įvykį naudojant Q vietoje P. Todėl kryžminės entropijos nuostolių sumažinimas padeda modeliui perkelti išvesties pasiskirstymą į tikslo funkciją. 2.10 lygtis yra matematinis kryžminės entropijos nuostolių  $L$  apibrėžimas, kur  $p(y_i)$  yra tikrosios klasės (angl. *ground truth*)  $y_i$  tikimybė. Apskaičiuojant duoto vaizdo tikimybę, iš tinklo išėjimų  $z$ , naudojama softmax funkcija (žr. 2.11 formulę). Didesnė tikros klasės tikimybė mažina kryžminės entropijos nuostolius. Semantinės segmentacijos kontekste, norint klasifikuoti kiekvieną pikselį, paprastai apskaičiuojama kiekvieno pikselio tikrosios klasės tikimybė ir naudojama kryžminės entropijos nuostolių funkcija, modelio parametrų optimizavimui [61].

$$p(y_i) = \frac{\exp(z_i)}{\sum_j \exp(z_j)} \quad (2.10)$$

$$L = \sum_{i=1}^n y_i \log(p(y_i)) \quad (2.11)$$

### 3. Tyrimų rezultatai ir jų aptarimas

Magistro baigiamajame projekte nagrinėtos 2 užduotys – vaizdų klasifikavimas į įtrūkusius ir sveikus bei įtrūkimų segmentavimas. Sprendžiant vaizdų klasifikavimo užduotį išbandytos EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB3 architektūros, apmokymo metu tiek naudojant, tiek nenaudojant vaizdų augmentaciją. Vėliau ieškant geriausio klasifikatoriaus išbandyti ir AlexNet bei VGG-16 modeliai, be vaizdų augmentacijos.

Vaizdai segmentuoti dviem būdais. Pirmas, kada vaizdas skaidomas dalimis po  $128 \times 128$  pikselius ir naudojant geriausiai klasifikavimo užduotį atlikusį modelį, o vėliau ir 3 geriausių modelių ansamblį, kiekvienoje iš jų skaičiuojama tikimybė, jog šioje vietoje yra įtrūkimas. Antru būdu naudota U-Net architektūra su EfficientNetB0, EfficientNetB1, EfficientNetB2, EfficientNetB3, EfficientNetB4 ir EfficientNetB5 pagrindiniais modeliais.

#### 3.1. Klasifikavimas

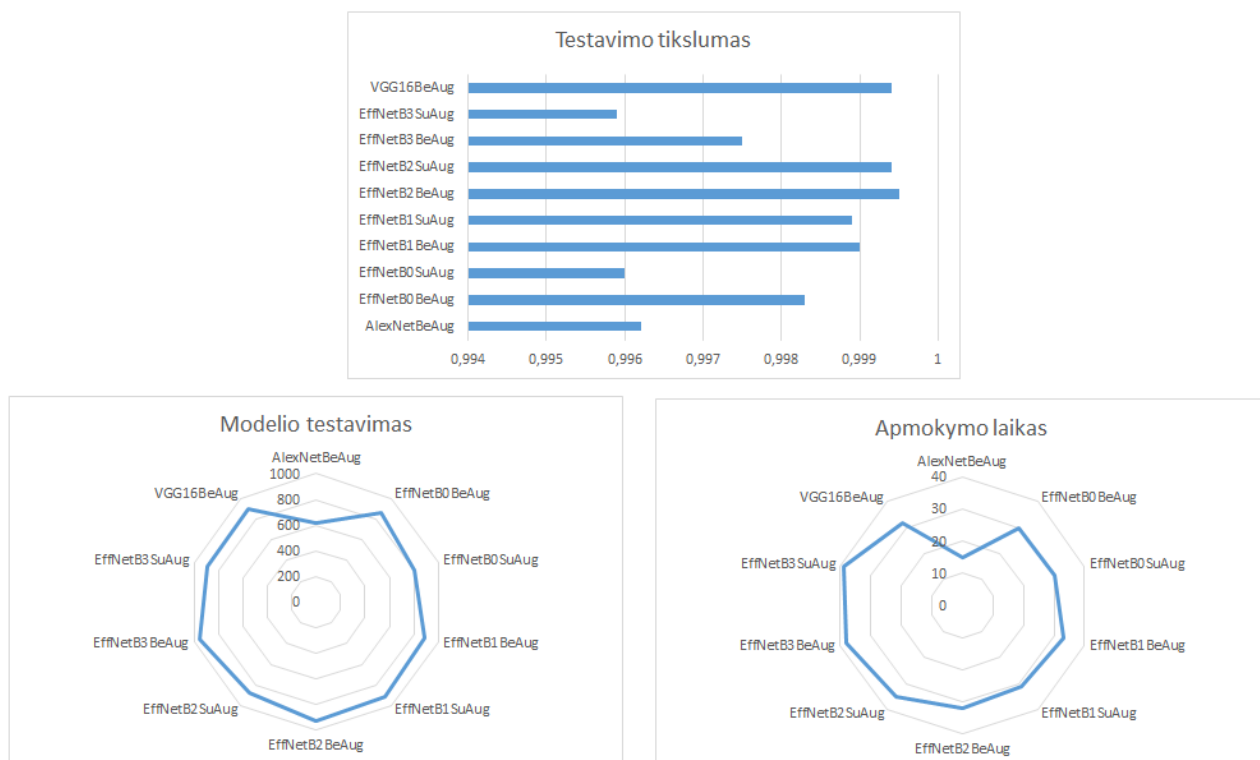
Magistro baigiamajame projekte sprendžiama 2 klasių klasifikavimo užduotis – vaizdas arba turi arba neturi įtrūkimo. Ieškant geriausiai šią užduotį atliekančio modelio vertintas tikslumas gautas testavimo imtyje, kiek laiko su turimais skaičiavimo resursais užtruko apmokyti modelį, per kiek epochų apmokytas modelis ir vėliau, po apmokymo, su visais modeliais bandyta klasifikuoti 1000 didelio formato, iki tol apmokyme, validacijoje ir testavime nenaudotų vaizdų. Vertinta kokią dalį iš šių vaizdų modelis klasifikavo teisingai. Rezultatai pateikti 3 lentelėje ir 16 paveiksle.

**3 lentelė.** Klasifikavimo modelių rezultatai

Modelis	Įvesties formatas	Testavimo tikslumas I <sup>1</sup>	Testavimo tilslumas II <sup>2</sup>	Epochų skaičius	Apmokymo laikas, min
EffNetB0BeAug	256 × 256	0,9983	0,855	30	30
EffNetB0SuAug	256 × 256	0,996	0,809	30	30
EffNetB1BeAug	256 × 256	0,999	0,892	30	33
EffNetB1SuAug	256 × 256	0,9989	0,914	30	31
EffNetB2BeAug	256 × 256	0,9995	0,931	30	32
EffNetB2SuAug	256 × 256	0,9994	0,879	30	35
EffNetB3BeAug	256 × 256	0,9975	0,952	30	38
EffNetB3SuAug	256 × 256	0,9959	0,890	30	39
AlexNetBeAug	128 × 128	0,9962	0,614	30	15
VGG16BeAug	256 × 256	0,9994	0,895	30	32

<sup>1</sup> Surface Crack Detection duomenys

<sup>2</sup> Nepriklausoma testavimo imtis su didesne raiška.



16 pav. Klasifikavimo modelių rezultatai

Gavus rezultatus pastebėta, jog modelių tikslumas vertinant testavimo imtyje labai artimas – didžiausias tikslumas nuo mažiausio skiriasi per 0,0036. Todėl nuspręsta laikyti, jog testavimo imtyje visų modelių tikslumas vienodas. Apmokymo laikas visiems modeliams taipogi gan artimas, išskyrus AlexNet. Kadangi šis tinklas mažiausias iš visų tyrinėtų ir turi mažiausiai parametru, atitinkamai jo apmokymo laikas ženkliai trumpesnis negu kitų. Dėl to ieškant geriausio modelio pagrindiniu kriterijumi laikyta teisingai klasifikuotų, didelio formato, vaizdų dalis. Daugiausiai vaizdų teisingai klasifikuota su EfficientNetB3 modeliu, kai apmokymo metu nenaudota vaizdų augmentacija. Teisingai klasifikuoti 952 vaizdai iš 1000. Antroje vietoje EfficientNetB2 modelis, be vaizdų augmentacijos. Teisingai klasifikuoti 931 vaizdas iš 1000. Trečioje vietoje EfficientNetB1 modelis, su vaizdų augmentacija. Teisingai klasifikuoti 914 vaizdų iš 1000.

Taipogi pastebėta, jog modeliai dažniausiai klaidingai klasifikuoja didelės raiškos sutrupėjusio betono vaizdus (žr. 17 pav.). Taip yra dėl to, jog apmokymui nenaudoti sutrupėjusio betono vaizdai. Dėl šios priežasties modeliai linkę sutrupėjusį betoną klasifikuoti kaip sveiką.

Trys geriausi modeliai vėliau sujungti į ansamblį ir naudoti įtrūkimų segmentavimui. Pateikiama detalesnė kiekvieno iš jų apmokymo informacija.



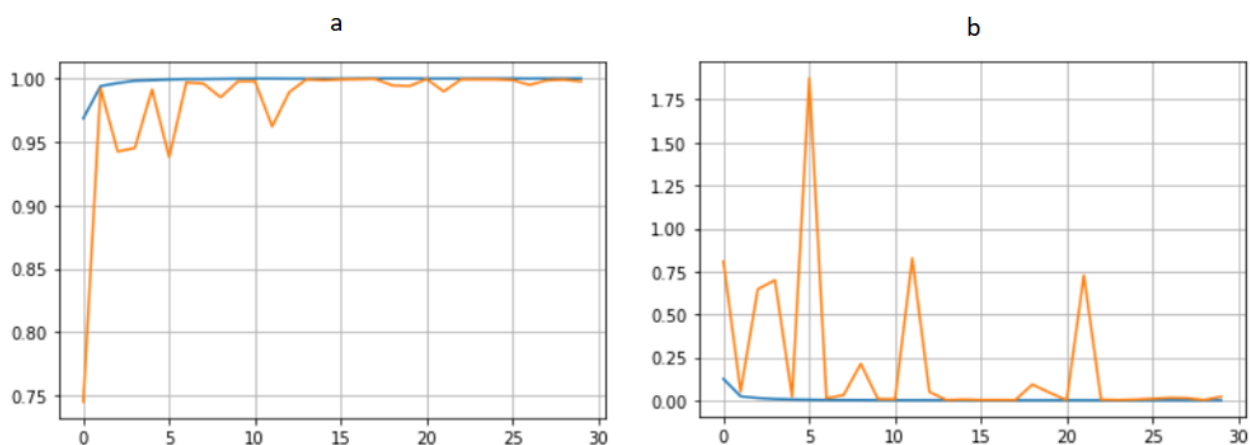
17 pav. Sutrupėjusio betono pavyzdys

### 3.1.1. EfficientNetB3 be augmentacijos

4 lentelėje pateikiami EfficientNetB3, be vaizdų augmentacijos, modelio testavimo imties klasifikavimo rezultatai. Matoma, jog modelis labiau linkęs klaidingai klasifikuoti sveiko betono vaizdą. Taipogi 18 paveiksle pateikiamas tikslumo ir nuostolių funkcijos reikšmės kitimas, apmokymo (mėlyna) ir testavimo (geltona) imtyse, apmokymo metu.

**4 lentelė.** EfficientNetB3 be augmentacijos modelio klasifikavimo rezultatai

<b>Progozė</b>	<b>Be įtrūkimo</b>	<b>Su įtrūkimu</b>
<b>Tikra reikšmė</b>		
Be įtrūkimo	3980	19
Su įtrūkimu	1	4000



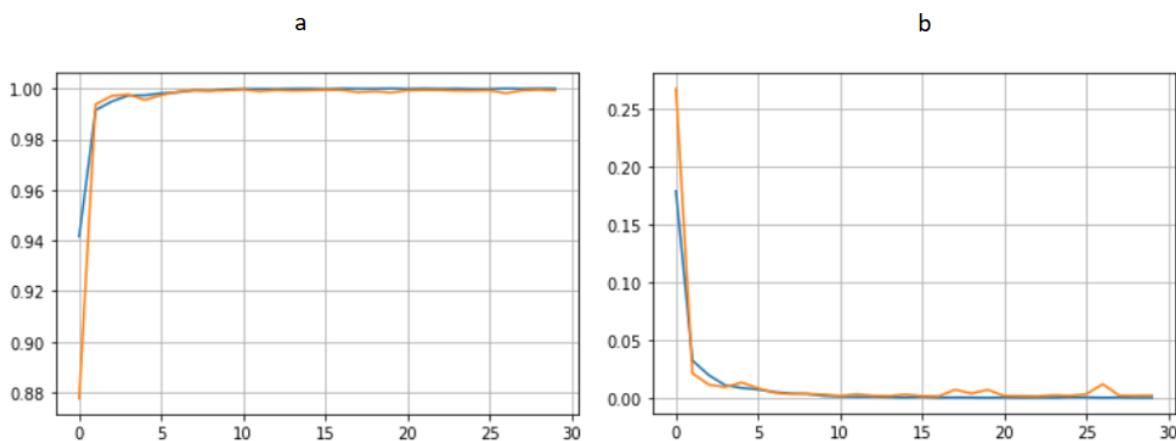
18 pav. EfficientNetB3 be augmentacijos modelio tikslumo a ir nuostolių funkcijos reikšmės b kitimas apmokymo metu

### 3.1.2. EfficientNetB2 be augmentacijos

5 lentelėje pateikiami EfficientNetB2, be vaizdų augmentacijos, modelio testavimo imties klasifikavimo rezultatai. Matoma, jog modelis itin tiksliai klasifikuoja vaizdus testavimo imtyje ir vienodai klaidingai prognozuoja įtrūkusio ir sveiko betono vaizdą. Taipogi 19 paveiksle pateikiamas tikslumo ir nuostolių funkcijos reikšmės kitimas, apmokymo (mėlyna) ir testavimo (geltona) imtyse, apmokymo metu.

5 lentelė. EfficientNetB3 be augmentacijos modelio klasifikavimo rezultatai

Prognozė \ Tikra reikšmė	Be įtrūkimo	Su įtrūkimu
Be įtrūkimo	3997	2
Su įtrūkimu	2	3999



19 pav. EfficientNetB2 be augmentacijos modelio tikslumo a ir nuostolių funkcijos reikšmės b kitimas apmokymo metu

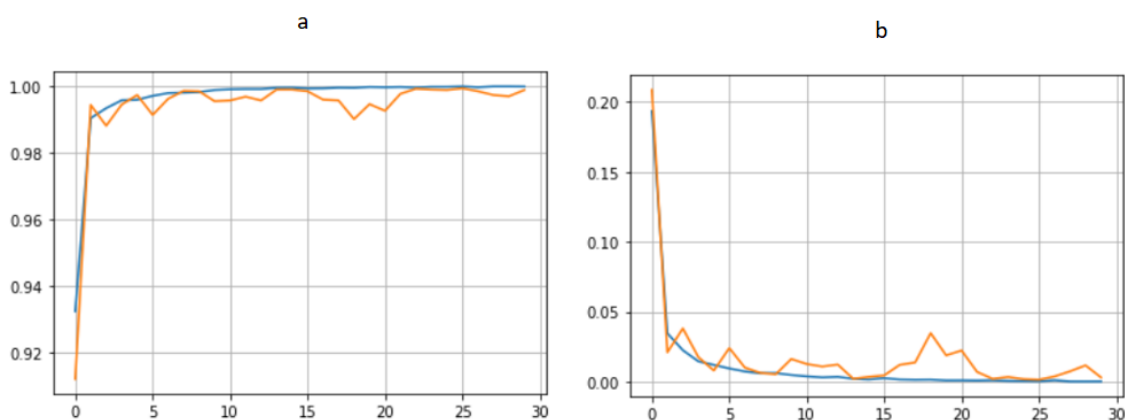


### 3.1.3. EfficientNetB1 su augmentacija

6 lentelėje pateikiami EfficientNetB1, su vaizdų augmentacija, modelio testavimo imties klasifikavimo rezultatai. Matoma, jog modelis itin tiksliai klasifikuoja vaizdus testavimo imtyje ir vienodai klaidingai prognozuoja įtrūkusio ir sveiko betono vaizdą. Taipogi 20 paveiksle pateikiamas tikslumo ir nuostolių funkcijos reikšmės kitimas, apmokymo (mėlyna) ir testavimo (geltona) imtyse, apmokymo metu.

6 lentelė. EfficientNetB1 su augmentacija modelio klasifikavimo rezultatai

Progozė \ Tikra reikšmė	Be įtrūkimo	Su įtrūkimu
Be įtrūkimo	3986	13
Su įtrūkimu	1	4000



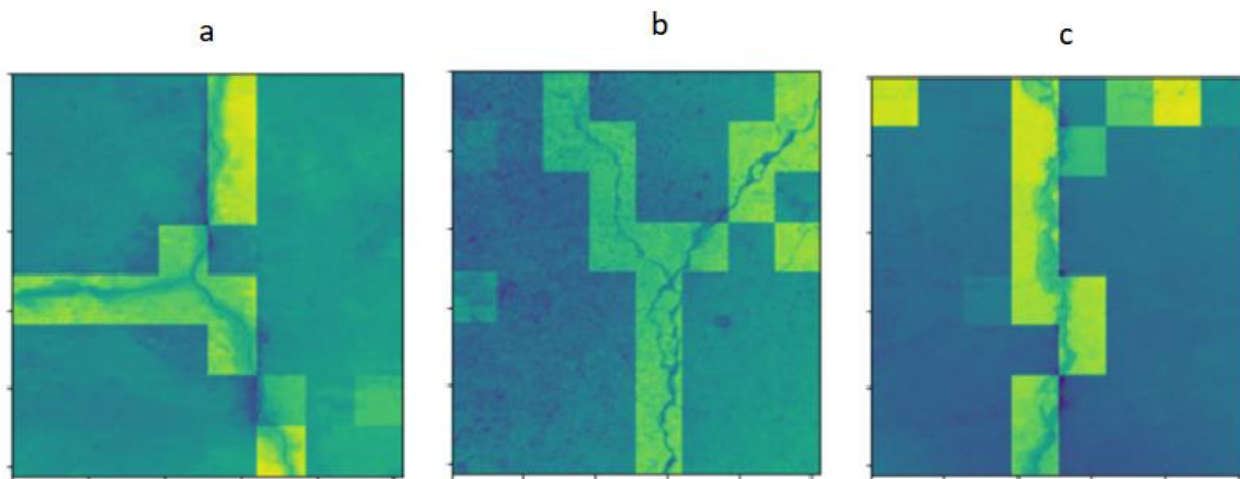
20 pav. EfficientNetB1 su augmentacija modelio tikslumo a ir nuostolių funkcijos reikšmės b kitimas apmokymo metu

### 3.2. Segmentavimas

Magistro baigiamajame projekte įtrūkimų segmentavimas atliktas dviem būdais – skaidant vaizdą į mažesnes dalis ir jose skaičiuojant įtrūkimo tikimybę, bei naudojant U-Net architektūrą. Pirmuoju būdu segmentavimas atliktas naudojant EfficientNetB3 modelį, be vaizdų augmentacijos ir modelių ansamblių, sudarytą iš EfficientNetB3, be vaizdų augmentacijos, EfficientNetB2, be vaizdų augmentacijos ir EfficientNetB1 su vaizdų augmentacija. Atliekant semantinį segmentavimą ieškota geriausio U-Net architektūros modelio-pagrindo.

### 3.2.1. Segmentavimas su EfficientNetB3, be vaizdų augmentacijos

Segmentavimas atliktas su EfficientNetB3 modeliu, be vaizdų augmentacijos, kuris prieš tai naudotas sprendžiant klasifikavimo užduotį. Šis metodas išbandytas su didelės raiškos,  $1024 \times 1024$  formato vaizdais. Kiekvienas iš jų suskaidytas į  $128 \times 128$  formato mažesnes sritis ir kiekvienoje iš jų apskaičiuota tikimybė, jog yra įtrūkimas. Gavus tikimybių matricą, ji išdidinta ir atvaizduota sugretinant su pradiniu vaizdu (žr. 21 pav.).

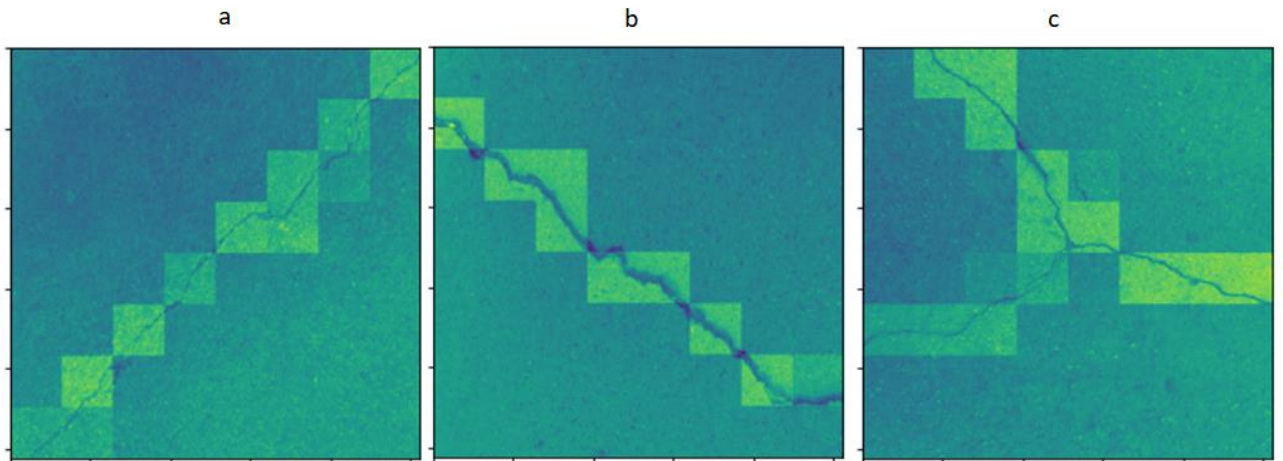


21 pav. Segmentuotų vaizdų, su EfficientNetB3 modeliu, be vaizdų augmentacijos, pavyzdys

Metodas greitai ir kokybiškai, be papildomo apmokymo, segmentavo įtrūkimus vaizduose, kurie iki tol nenaudoti apmokyme, validacijoje arba testavime. Pasitaikė sričių, kuriose pažymėtas ne visas įtrūkimas (žr. 21 pav. c). Taip daugiausiai nutiko dėl to, jog įtrūkimas eina 2 zonų sandūra. Taipogi pasitaikė ir klaidingai klasifikuotų sričių (žr. 21 pav. c). Tačiau nė vienoje išbandytoje nuotraukoje tokių zonų nebuvo daugiau negu 7, iš vaizde esančių 64.

### 3.2.2. Segmentavimas su modelių ansambliu

Sujungus tris geriausius klasifikavimo modelius, EfficientNetB3, be vaizdų augmentacijos, EfficientNetB2, be vaizdų augmentacijos ir EfficientNetB1 su vaizdų augmentacija, gautas ansamblis. Jį naudojant įtrūkimų segmentavimui, kiekvienu iš jų apskaičiuojama tikimybė, kad suskaidyto vaizdo zonoje yra įtrūkimas ir randamas šių tikimybių vidurkis. Gautas vidurkis išsaugomas matricoje, kuri vėliau išdidinama ir karščio žemėlapiu pavidalu atvaizduojama ant pradinio vaizdo (žr. 22 pav.).

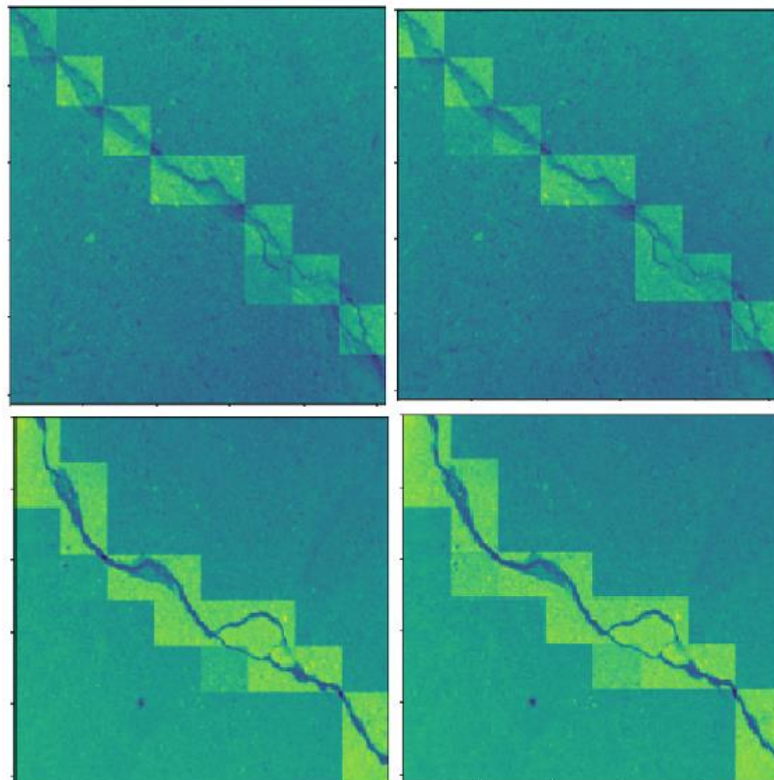


22 pav. Segmentuotų vaizdų, su modelių ansambliu, pavyzdys

Modelių ansambliu pavyko gauti tikslius rezultatus. Visais išbandytais atvejais įtrūkimai pažymėti teisingai. Pasitaikė įtrūkimo dalių, kurių ansamblis neįvardino kaip įtrūkimo, kadangi jis užėmė labai mažą zonos dalį, arba ėjo zonų sandūra (žr. 22 pav.).

### 3.2.3. Segmentavimo su EfficientNetB3, be vaizdų augmentacijos, modelio ir ansamblio rezultatų palyginimas

23 paveiksle pateikiami segmentavimo rezultatai tai pačiai nuotraukai, gauti su EfficientNetB3, be vaizdų augmentacijos, modeliu (kairėje) ir modelių ansambliu (dešinėje).



23 pav. Segmentavimo rezultatai su EfficientNetB3, be vaizdų augmentacijos ir su modelių ansambliu

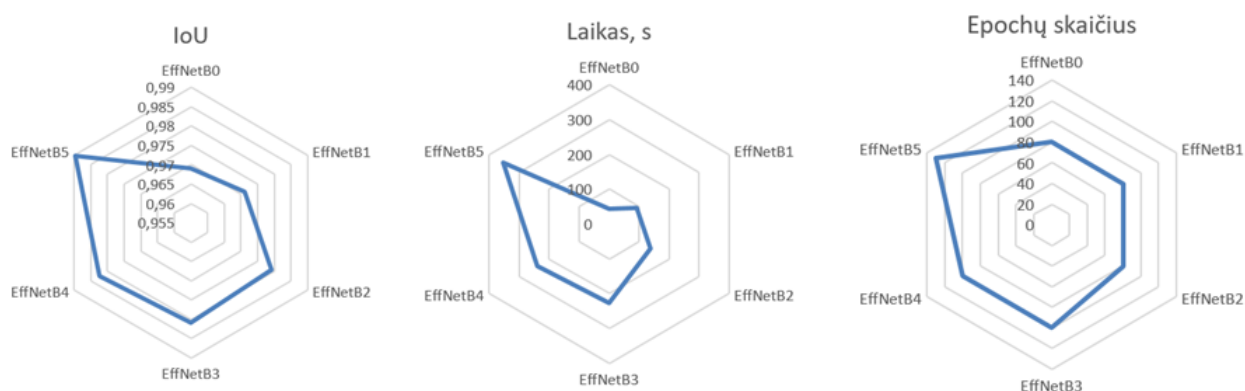
Atlikus bandymus su skirtingais vaizdais didesnių skirtumų tarp rezultatų neįžvelgta. Modelių ansamblis kai kada užtikrinčiau pažymi zoną kaip su galimu įtrūkimu, tačiau EfficientNetB3 modelis, be vaizdų augmentacijos, visais atvejais tas pačias vietas irgi išskirdavo kaip galimą įtrūkimą. Kadangi tiek modelių ansamblis, tiek EfficientNetB3, be vaizdų augmentacijos, modelis vienodai kokybiškai atliko įtrūkimų segmentavimo užduotį, laikoma, kad atliekant šią užduotį pakanka vieno, geriausio, EfficientNetB3, be vaizdų augmentacijos, modelio. Naudojant tik jį pasiekiami panašūs rezultatai, tačiau nereikia apmokyti kitų modelių, kam reikalingi skaičiavimo ir laiko resursai.

### 3.2.4. Segmentavimas naudojant U-Net

Kitu etapu išbandytas semantinis įtrūkimų segmentavimas, naudojant U-Net tipo architektūrą. Baigiamajame projekte išbandytos skirtingos U-Net modifikacijos, kada keičiamas modelio pagrindas. Kiekvieną kartą apmokymui naudota 100 atsitiktinai atrinktų vaizdų su įtrūkimais ir jų kaukėmis iš „Surface Crack Detection“ duomenų rinkinio. Rezultatai pateikiami 7 lentelėje ir 24 paveiksle.

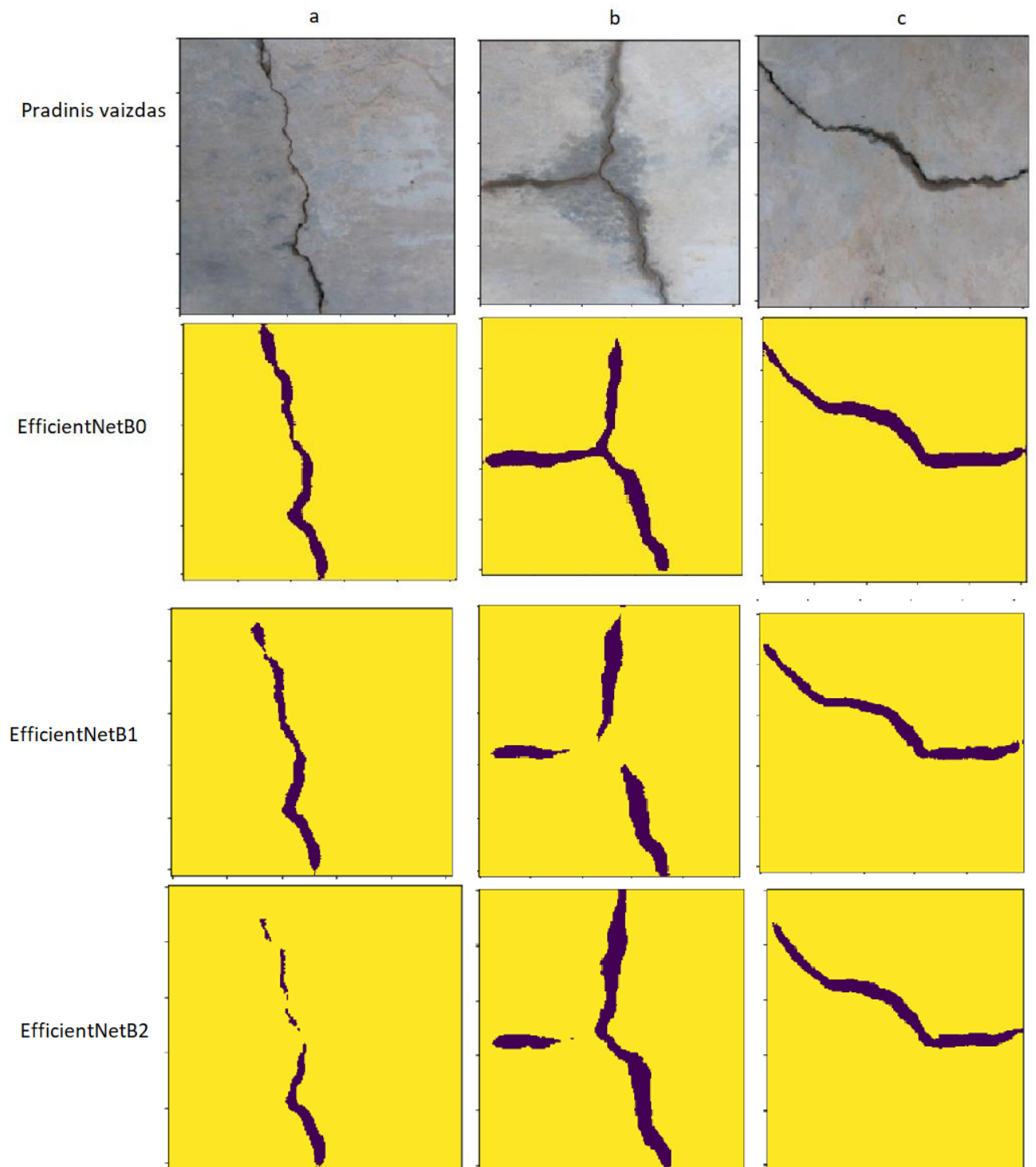
7 lentelė. U-Net rezultatai

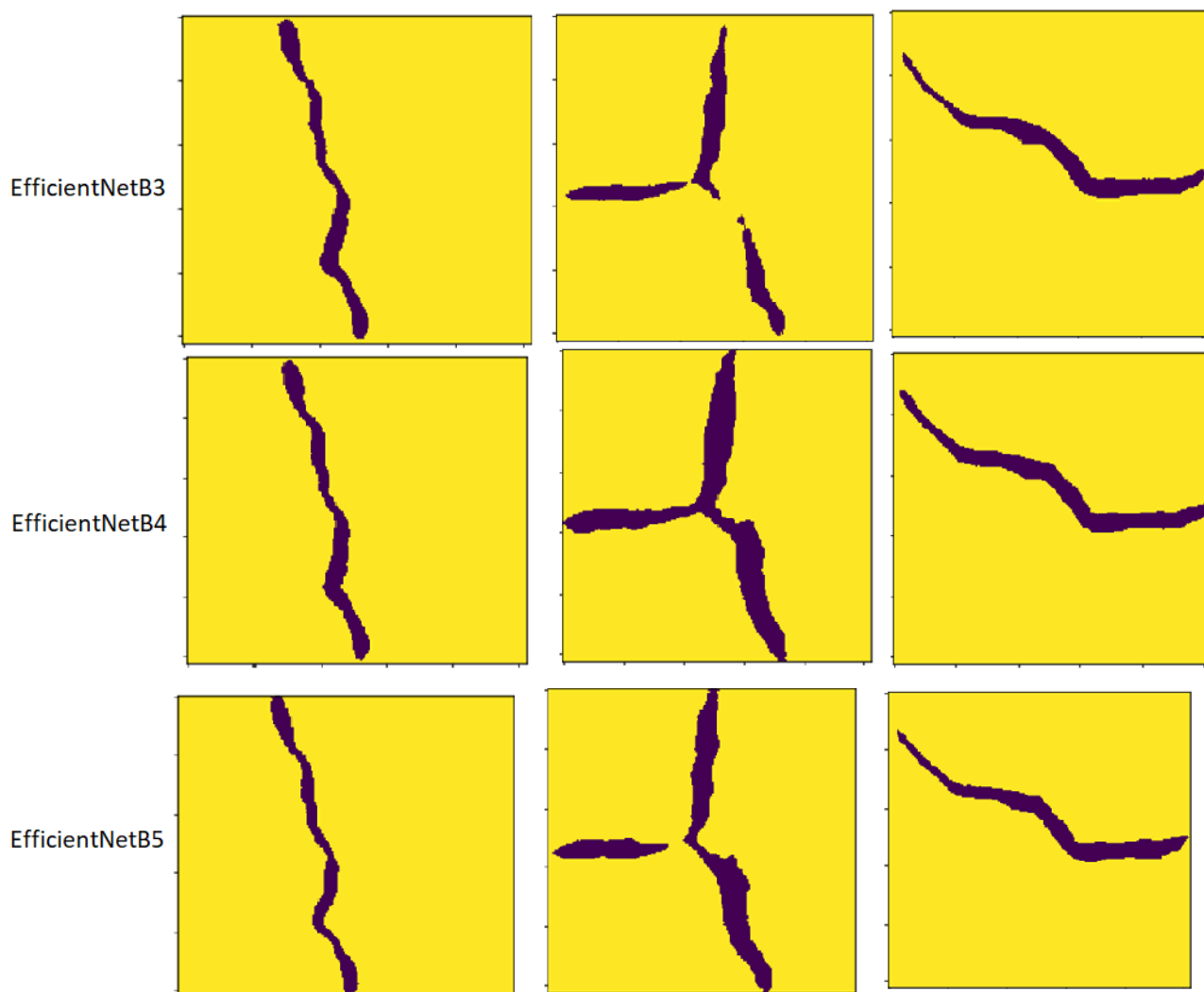
Modelis	IoU	Epochų skaičius	Apmokymo laikas, s
EffNetB0	0,969	80	44
EffNetB1	0,9711	80	92
EffNetB2	0,9792	80	139
EffNetB3	0,9809	100	227
EffNetB4	0,9824	100	240
EffNetB5	0,9897	130	352



24 pav. U-Net rezultatai

U-Net modelis, su skirtingai pagrindais, net ir naudojant nedidelę apmokymo imtį, rodo gerus rezultatus. Kadangi IoU reikšmės visais atvejais labai artimos (didžiausia nuo mažiausios skiriasi per 0,0207), o apmokymo laikas nė su vienu modeliu neviršijo 6 minučių, ieškant geriausios architektūros semantiniam segmentavimui nuspręsta išbandyti kiekvieną atvejį su didelio formato vaizdais.





25 pav. Segmentavimo su U-Net pavyzdžiai

Matoma, kad visi modeliai pakankamai sėkmingai atliko užduotį. Pirmą nuotrauką (žr. 25 pav. a) geriausiai segmentavo EfficientNetB0. Šis įtrūkimą segmentavo ploniau nei kiti modeliai, o EfficientNetB2 dalies vaizdo nenurodė kaip su įtrūkimu. Antrą vaizdą (žr. 25 pav. b) taipogi geriausiai segmentavo EfficientNetB0. EfficientNetB1, EfficientNetB2, EfficientNetB3 ir EfficientNetB5 apskritai dalies įtrūkimo nesegmentavo. EfficientNetB4 modelis segmentavo visą įtrūkimą, tačiau jis gavosi platesnis nei yra iš tiesų. Dėl šių priežasčių laikoma, kad antrą vaizdą geriausiai segmentavo EfficientNetB0 modelis. Paskutinį vaizdą (žr. 25 pav. c) visi modeliai segmentavo pakankamai kokybiškai. Visgi, kiek ploniau įtrūkimą segmentavo EfficientNetB1 modelis, todėl laikoma, kad šiam vaizdai jo rezultatas artimiausias tikrajam.

Išbandžius modelius su dar keletu atsitiktinai atrinktų vaizdų vyravo ta pati tendencija, kad EfficientNetB0 modelis įtrūkimus segmentuoja kiek plonesnius, taip pateikdamas tiksliausius rezultatus. Tad laikoma, kad geriausias modelis įtrūkimų semantiniam segmentavimui yra su EfficientNetB0 pagrindu.

## Išvados

1. Atlikus literatūros apžvalgą sužinota, kad įtrūkimų aptikimui betono konstrukcijose naudojami metodai grįsti giliuoju mokymusi, arba papildomais įrankiais, kurie seka pokyčius betone.
2. Geriausias „Surface Crack Detection“ imties betono vaizdų klasifikavimo modelis – EfficientNetB3 be vaizdų augmentacijos. Testavimo imties tikslumas 0,9975; teisingai klasifikuotų nepriklausomos imties nuotraukų dalis 0,952; mokymosi laikas – 38 min.
3. Vertinant vizualiai, grubaus segmentavimo metodas, grįstas klasifikatoriaus tikimybėmis, gan korektiškai diferencijuoja trūkio pikselius nuo sveiko betono pikselių. Klasifikavimo modelių ansamblis nedavė ženkliai geresnio rezultato aptinkant trūkio pikselius.
4. Semantiniame segmentavime pavyko pasiekti gerus rezultatus su nedidele apmokymo imtimi. Pasiektas IoU = 0,9897 naudojant tik 100 vaizdų imtį. Galima daryti išvadą, jog metodas yra tinkamas įgyvendinti realiomis sąlygomis betono trūkių detekcijai ir segmentavimui.



## Literatūros sąrašas

1. Sparavigna, Amelia Carolina. *Ancient Concrete Works*. 2011.
2. Azenha, Miguel, and Rui Faria. "Temperatures and Stresses Due to Cement Hydration on the R/C Foundation of a Wind Tower—A Case Study." *Engineering Structures*, t. 30, nr. 9, p. 2392–400, rugsėjis 2008.
3. Kovári, Kalman. "History of the Sprayed Concrete Lining Method—Part I: Milestones up to the 1960s." *Tunnelling and Underground Space Technology*, t. 18, nr. 1, vas. 2003, p. 57–69
4. pucolanai. <https://www.vle.lt/straipsnis/pucolanai/> (žiūrėta kovo 7, 2022).
5. Lancaster, Lynne C. *Concrete Vaulted Construction in Imperial Rome: Innovations in Context*. Cambridge University Press, 2005.
6. T., Arun Prasath, et al. "Mechanical Properties of Concrete Beam with Steel Fiber and Bauxite Residual." *Materials Today: Proceedings*, t. 37, 2021, p. 3359–62.
7. Nishikawa, Takafumi, et al. "Concrete Crack Detection by Multiple Sequential Image Filtering: Concrete Crack Detection by Image Processing." *Computer-Aided Civil and Infrastructure Engineering*, t. 27, nr. 1, saus. 2012, p. 29–47.
8. Beiser, Vince. "Florida's Condo Collapse Foreshadows the Concrete Crack-Up." *Wired*. [www.wired.com, https://www.wired.com/story/floridas-condo-collapse-foreshadows-the-concrete-crack-up/](https://www.wired.com/story/floridas-condo-collapse-foreshadows-the-concrete-crack-up/) (žiūrėta kovo 28, 2022).
9. Ye, Wenlong, et al. "Deep Learning-Based Fast Detection of Apparent Concrete Crack in Slab Tracks with Dilated Convolution." *Construction and Building Materials*, t. 329, bal. 2022, p. 127157.
10. ASCE's 2017 infrastructure report card: bridges. Reston: American Society of Civil Engineers.
11. Zhang, Qianyun, et al. "Real-Time Detection of Cracks on Concrete Bridge Decks Using Deep Learning in the Frequency Domain." *Engineering*, t. 7, nr. 12, gruod. 2021, p. 1786–96.
12. Miao, Pengyong, and Teeranai Srimahachota. "Cost-Effective System for Detection and Quantification of Concrete Surface Cracks by Combination of Convolutional Neural Network and Image Processing Techniques." *Construction and Building Materials*, t. 293, liepos 2021, p. 123549.
13. Yang, Yuan-Sen, et al. "Thin Crack Observation in a Reinforced Concrete Bridge Pier Test Using Image Processing and Analysis." *Advances in Engineering Software*, t. 83, geg. 2015, p. 99–108.
14. Abdel-Qader, Ikhlas, et al. "PCA-Based Algorithm for Unsupervised Bridge Crack Detection." *Advances in Engineering Software*, t. 37, nr. 12, gruod. 2006, p. 771–78.
15. Ronneberger, Olaf, et al. "U-Net: Convolutional Networks for Biomedical Image Segmentation." *ArXiv:1505.04597* Cs, geg. 2015. [Interaktyvus]. Adresas: [arXiv.org, http://arxiv.org/abs/1505.04597](http://arxiv.org/abs/1505.04597).
16. Long, Jonathan, et al. "Fully Convolutional Networks for Semantic Segmentation." *ArXiv:1411.4038* Cs, kovo 2015. [Interaktyvus]. Adresas: [arXiv.org, http://arxiv.org/abs/1411.4038](http://arxiv.org/abs/1411.4038).
17. Liu, Zhenqing, et al. "Computer Vision-Based Concrete Crack Detection Using U-Net Fully Convolutional Networks." *Automation in Construction*, t. 104, rugpjūčio. 2019, p. 129–39.



18. Markets and Markets, Structural Health Monitoring Market by Technology (Wired and Wireless), Offering (Hardware (Sensors, Data Acquisition Systems) and Software & Services), Vertical (Civil Infrastructure, Energy), Application, and Geography – Global Forecast to 2023, Markets and Markets, 2019
19. Park, Song Ee, et al. “Concrete Crack Detection and Quantification Using Deep Learning and Structured Light.” *Construction and Building Materials*, t. 252, regpjūčio. 2020, p. 119096.
20. Wang, Tai-Tien. “Characterizing Crack Patterns on Tunnel Linings Associated with Shear Deformation Induced by Instability of Neighboring Slopes.” *Engineering Geology*, t. 115, nr. 1–2, rugs. 2010, p. 80–95.
21. Liu, Baoju, and Tengyu Yang. “Image Analysis for Detection of Bugholes on Concrete Surface.” *Construction and Building Materials*, t. 137, bal. 2017, p. 432–40.
22. Nhat-Duc, Hoang, et al. “Automatic Recognition of Asphalt Pavement Cracks Using Metaheuristic Optimized Edge Detection Algorithms and Convolution Neural Network.” *Automation in Construction*, t. 94, spal. 2018, p. 203–13.
23. Ren, Yupeng, et al. “Image-Based Concrete Crack Detection in Tunnels Using Deep Fully Convolutional Networks.” *Construction and Building Materials*, t. 234, Feb. 2020, p. 117367.
24. Yu, Fisher, and Vladlen Koltun. “Multi-Scale Context Aggregation by Dilated Convolutions.” *ArXiv:1511.07122 Cs*, bal. 2016. [Interaktyvus]. Adresas: [arXiv.org http://arxiv.org/abs/1511.07122](http://arxiv.org/abs/1511.07122).
25. Dekelbab, Waseem, Adel Al-Wazeer, and Bobby Harris. "History lessons from the national bridge inventory." *Public Roads* 71.6 (2008): 30.
26. Dorafshan, Sattar, et al. "Fatigue crack detection using unmanned aerial systems in under-bridge inspection." (2017): 1.
27. Gucunski, Nenad, et al. “Delamination and Concrete Quality Assessment of Concrete Bridge Decks Using a Fully Autonomous RABIT Platform.” *Structural Monitoring and Maintenance*, t. 2, nr. 1, kovo 2015, p. 19–34.
28. Lim, Ronny Salim, et al. “A Robotic Crack Inspection and Mapping System for Bridge Deck Maintenance.” *IEEE Transactions on Automation Science and Engineering*, t. 11, nr. 2, bal. 2014, p. 367–78.
29. Dorafshan, Sattar, et al. “Comparison of Deep Convolutional Neural Networks and Edge Detectors for Image-Based Crack Detection in Concrete.” *Construction and Building Materials*, t. 186, spal. 2018, p. 1031–45.
30. ONDRAF/NIRAS, The Long-Term Safety Strategy for the Geological Disposal of Radioactive Waste, ONDRAF/NIRAS report NIROND-TR 2009-14 E (2009)
31. Bennett, D. G., and R. Gens. “Overview of European Concepts for High-Level Waste and Spent Fuel Disposal with Special Reference Waste Container Corrosion.” *Journal of Nuclear Materials*, t. 379, nr. 1–3, rugs. 2008, p. 1–8.
32. Féron, Damien, et al. “Corrosion Issues in Nuclear Waste Disposal.” *Journal of Nuclear Materials*, t. 379, nr. 1–3, rugs. 2008, p. 16–23.
33. Iliopoulos, S., et al. “Detection and Evaluation of Cracks in the Concrete Buffer of the Belgian Nuclear Waste Container Using Combined NDT Techniques.” *Construction and Building Materials*, t. 78, kovo. 2015, p. 369–78.

34. Schwartz, Eric, et al. "The Use of Digital Image Correlation for Non-Destructive and Multi-Scale Damage Quantification." *Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems 2013*, t. 8692, 2013, p. 706–20.
35. Corr, David, et al. "Digital Image Correlation Analysis of Interfacial Debonding Properties and Fracture Behavior in Concrete." *Engineering Fracture Mechanics*, t. 74, nr. 1–2, sausio. 2007, p. 109–21.
36. Grosse, Christian U., et al., eds. *Acoustic Emission Testing: Basics for Research-Applications in Engineering*. Springer Nature, 2021.
37. Naik, Tarun R., V. Mohan Malhotra, and John S. Popovics. "The ultrasonic pulse velocity method." *Handbook on Nondestructive Testing of Concrete*, Second Edition. CRC Press, 2003. 8-1.
38. Wang, Fangying, et al. "Testing and Numerical Modelling of Circular CFDST Cross-Sections with Stainless Steel Outer Tubes in Bending." *Engineering Structures*, t. 247, lapk. 2021, p. 113170.
39. Bungey, John H., and Michael G. Grantham. *Testing of concrete in structures*. Crc Press, 2006.
40. Viswanath, Somashekar, et al. "Concrete Compressive Strain Behavior and Magnitudes under Uniaxial Fatigue Loading." *Construction and Building Materials*, t. 296, rugpj. 2021, p. 123718.
41. Guzmán-Torres, José A., et al. "Effective Medium Crack Classification on Laboratory Concrete Specimens via Competitive Machine Learning." *Structures*, t. 37, kovo. 2022, p. 858–70.
42. Geng, Zhiqiang, et al. "Novel Target Attention Convolutional Neural Network for Relation Classification." *Information Sciences*, t. 597, birž. 2022, p. 24–37.
43. Valueva, M. V., et al. "Application of the Residue Number System to Reduce Hardware Costs of the Convolutional Neural Network Implementation." *Mathematics and Computers in Simulation*, t. 177, lapkr. 2020, p. 232–43.
44. What Are Convolutional Neural Networks? <https://www.ibm.com/cloud/learn/convolutional-neural-networks> (žiūrėta balandžio 20, 2022).
45. <http://metacoder.ai/blog/2021/12/13/the-top-12-activation-functions-a-short-guide-for-choosing/>. (žiūrėta balandžio 20, 2022).
46. "Explain Pooling Layers: Max Pooling, Average Pooling, Global Average Pooling, and Global Max Pooling." *Knowledge Transfer*, <https://androidkt.com/explain-pooling-layers-max-pooling-average-pooling-global-average-pooling-and-global-max-pooling/> (žiūrėta balandžio 20, 2022)..
47. Saha, Sumit. "A Comprehensive Guide to Convolutional Neural Networks — the ELI5 Way." *Medium*, 17 gruod. 2018. [Interaktyvus]. Adresas: <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>.
48. Vijayalakshmi, Kaliyamoorthy, et al. "Prediction of Virtual Energy Storage Capacity of the Air-Conditioner Using a Stochastic Gradient Descent Based Artificial Neural Network." *Electric Power Systems Research*, t. 208, liepos 2022, p. 107879.
49. Hou, Shaokang, and Yaoru Liu. "Early Warning of Tunnel Collapse Based on Adam-Optimised Long Short-Term Memory Network and TBM Operation Parameters." *Engineering Applications of Artificial Intelligence*, t. 112, birž. 2022, p. 104842.
50. R, Sakthivel, et al. "An Efficient Hardware Architecture Based on an Ensemble of Deep Learning Models for COVID -19 Prediction." *Sustainable Cities and Society*, t. 80, geg. 2022, p. 103713.

51. Shamila Ebenezer, A., et al. "Effect of Image Transformation on EfficientNet Model for COVID-19 CT Image Classification." *Materials Today: Proceedings*, t. 51, 2022, p. 2512–19.
52. Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." [Interaktyvus]. Adresas: arXiv preprint arXiv:1409.1556 (2014)
53. "VGG Very Deep Convolutional Networks (VGGNet) - What You Need to Know." Viso.Ai, 6 spal. 2021, <https://viso.ai/deep-learning/vgg-very-deep-convolutional-networks/>.
54. Dietlmeier, Julia, et al. "Few-Shot Hypercolumn-Based Mitochondria Segmentation in Cardiac and Outer Hair Cells in Focused Ion Beam-Scanning Electron Microscopy (FIB-SEM) Data." *Pattern Recognition Letters*, t. 128, gruod. 2019, p. 521–28.
55. Razavian, Ali Sharif, et al. "CNN Features Off-the-Shelf: An Astounding Baseline for Recognition." ArXiv:1403.6382 Cs, geg. 2014. [Interaktyvus]. Adresas: arXiv.org, <http://arxiv.org/abs/1403.6382>.
56. Le, Khuyen. "An Overview of VGG16 and NiN Models." *MLearning.Ai*, 8 gruod. 2021, <https://medium.com/mllearning-ai/an-overview-of-vgg16-and-nin-models-96e4bf398484>.
57. Krizhevsky, Alex, et al. "ImageNet Classification with Deep Convolutional Neural Networks." *Communications of the ACM*, t. 60, nr. 6, geg. 2017, p. 84–90.
58. Luo, Xianxian, et al. "Health Classification of Meibomian Gland Images Using Keratography 5M Based on AlexNet Model." *Computer Methods and Programs in Biomedicine*, t. 219, bir. 2022, p. 106742.
59. Pham, Van-Truong, et al. "EAR-UNet: A Deep Learning-Based Approach for Segmentation of Tympanic Membranes from Otoscopic Images." *Artificial Intelligence in Medicine*, t. 115, geg. 2021, p. 102065.
60. Kotu, Vijay, and Bala Deshpande. "Data Mining Process." *Predictive Analytics and Data Mining*, Elsevier, 2015, p. 17–36.
61. Dong, Yafen, et al. "Recognition of Imbalanced Underwater Acoustic Datasets with Exponentially Weighted Cross-Entropy Loss." *Applied Acoustics*, t. 174, kovo 2021, p. 107740.