

TALLINNA TEHNIKAÜLIKOOL
Infotehnoloogia teaduskond

Triin Rüütli 204280IABM

Metsloomade tuvastamine termopiltidelt kasutades sügavaid närvivõrke

Magistritöö

Juhendaja: Martin Rebane
MSc

Tallinn 2022

Autorideklaratsioon

Kinnitan, et olen koostanud antud lõputöö iseseisvalt ning seda ei ole kellegi teise poolt varem kaitsmisele esitatud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on töös viidatud.

Autor: Triin Rüütli

11.05.2022

Annotatsioon

Käesoleva töö eesmärk on töötada välja konvolutsioonilise närvivõrgu mudel, mis suudaks tuvastada metslooma objekti termopildilt. Töö käigus on loodud loomade andmekogu, mis koosneb metsloomadest jäädvustatud termopiltidest maanteeäärsetel aladel. Uurimustöö motivatsiooniks on parendada Eestis kasutusel olevat loomatuvaustusüsteemi ning suurendada selle töökindlust.

Metsloomade tuvastamiseks on kasutatud siirdeõppe meetodit mitme erineva konvolutsioonilise närvivõrguga. Objektide lokaliseerimiseks ning kategoriseerimiseks rakendati nii objektituvastuse mudeleid kui ka isendi segmenteerimise mudeleid. Saadud tulemuste osas on läbi viidud analüüs, mille raames uuritakse, milline närvivõrk sobib termopildilt objektide tuvastamiseks kõige paremini, kuidas mõjutavad erinevad ilmastikutingimused termopildilt objektituvastust ning kas tuvastamise täpsus langeb kaugemate objektide puhul.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 42 leheküljel, 6 peatükki, 34 joonist, 6 tabelit.

Abstract

Detecting Wild Animals From Thermal Images Using Deep Neural Networks

The purpose of this thesis is to develop a convolutional neural network that can detect wild animals from thermal images. In scope of the thesis, a dataset of animals has been created, which consists of thermal images of wild animals in roadside areas. The motivation of this research is to improve the animal detection system currently used in Estonia and to increase it's reliability.

Animals were detected by using transfer learning with multiple pre-trained convolutional neural networks. Object detection and instance segmentation models were used to locate and categorize objects in thermal images. The aim of analysing the results is to determine which architecture is best suited for detecting objects in a thermal image, how are different weather conditions affecting object recognition from thermal images and whether detection accuracy decreases for distant objects.

The thesis is in Estonian and contains 42 pages of text, 6 chapters, 34 figures, 6 tables.

Lühendite ja mõistete sõnastik

Epohh	<i>Epoch.</i> Närvivõrgu treenimisel üks treeningtsükkel kogu andmestikuga.
Isendi segmenteerimine	<i>Instance segmentation.</i> Tehisnägemise ülesanne, mis segmenteerib sisendpildil kõik otsitavad objektide eksemplarid.
Iteratsioon	<i>Iteration.</i> Närvivõrgu treenimisel kasutatav termin, mis näitab, mitu korda algoritmi parameetreid värskendatakse.
LTS	Loomatuvastussüsteem
Mudeli kaalud	<i>Model weights.</i> Parameetrite kogum, mis teisendab sisendandmeid närvivõrgu peidetud kihtides.
Närvivõrk	<i>Neural network.</i> Arvutuslik arhitektuur, mis võimaldab masinõppe algoritmide abil töödelda keerulisi andmestikke.
Põhitõe andmed	<i>Ground truth data.</i> Originaalandmed, mis on tuvastatavad otsese vaatluse, mitte kaugseire põhjal tehtud järelduste kaudu.
RGB	<i>Red-green-blue.</i> Värvimudel, milles esinevad kolm põhivärvi: punane (<i>red</i>), roheline (<i>green</i>), sinine (<i>blue</i>). Teised värvid tekivad nende kolme liitmisel.
Siirdeõpe	<i>Transfer learning.</i> Lähteülesandelt informatsiooni üle kandmine sihtülesande lahendamiseks.
Treeningpiltide partii	<i>Batch.</i> Närvivõrgu treenimiseks kasutatav parameeter, mis määrab, mitu pilti treenitakse ühe iteratsiooni jooksul.
Õpisamm	<i>Learning rate.</i> Närvivõrgu treenimisel kasutatav parameeter, mille eesmärk on optimeerida kaale nii, et viga oleks minimaalne.
YOLO	<i>You only look once.</i> Närvivõrk, mis suudab pildilt objekti tuvastada ühe etapiga.

Sisukord

1 Sissejuhatus	11
2 Ulukiõnnetused ning loomatu vastussüsteemid.....	13
2.1 Ulukiõnnetuste statistika.....	13
2.2 Metsloomade käitumismustrid ning nende seos õnnetustega.....	14
2.3 Loomatu vastussüsteemid.....	15
2.3.1 Eestis kasutusel olev loomatu vastussüsteem.....	15
2.3.2 Loomatu vastussüsteemide erinevad tehnoloogiad	19
2.4 Tehisnägemise alamvaldkonnad.....	21
2.4.1 Objektitu vastus	22
2.4.2 Pildi segmenteerimine	24
2.4.3 Mudeli tulemuste hindamine	25
3 Metoodika ja realisatsioon.....	28
3.1 Andmete kogumine ning puhastamine, andmekogu loomine	29
3.2 Andmete annoteerimine.....	30
3.3 Erinevate närvivõrkude rakendamine	31
3.3.1 Närvivõrkude treenimine.....	33
4 Mudelite treenimine ja valideerimine.....	36
4.1 Loomade termopiltide annoteeritud andmekogu loomine.....	37
4.2 Loomade tuvastamine segmenteerimise teel	39
4.2.1 Ühe kategooriaga annoteeritud andmestikul treenimine	39
4.2.2 Mitme kategooriaga annoteeritud andmestikul treenimine	41
4.3 Objektide tuvastamine piirdekasti meetodil	42
4.3.1 Fast R-CNN mudeli treenimine.....	42
4.3.2 YOLOv5 mudeli treenimine.....	44
5 Analüüs ja järeldused.....	46
5.1 Kuidas ning milliseid süvaõppe meetodeid kasutades saab tuvastada metsloomi termopiltidelt?.....	46
5.2 Kuidas mõjutavad erinevad ilmastikutingimused termopiltidelt objekti tuvastust?	47
5.3 Kas tuvastamise täpsus langeb kauguse suurenedes?.....	49
5.4 Ettepanekud loomatu vastussüsteemi parendamiseks	50
5.5 Edasised arendused.....	52

6 Kokkuvõte	53
Kasutatud kirjandus	54
Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks	57
Lisa 2 – märtsikuu termoandmetel rakendatud treenimata närvivõrgu tulemused.....	58
Lisa 3 – ühe kategooriaga annoteeritud andmekoguga treenitud Mask R-CNN mudeli tulemus videol	59
Lisa 4 – mitme kategooriaga annoteeritud andmekoguga treenitud Mask R-CNN mudeli tulemus videol	60
Lisa 5 – mitme kategooriaga annoteeritud andmekoguga treenitud YOLOv5 mudeli tulemus videol	61
Lisa 6 – erinevates ilmastikutingimustes Yolov5 mudeli rakendamine	62

Joonised

Joonis 1. Andmebaasi kantud ulukite liigiline koosseis. [2]	13
Joonis 2. Hukkunute ja vigastatute arv kokkupõrgetes loomadega. [3]	14
Joonis 3. Loomatuvastussüsteemide asukohad Eestis. [3]	16
Joonis 4. Radarite ning kaamerate paigutus. [3].....	17
Joonis 5. Loomatuvastussüsteemi alarmi jagunemine vastavalt põhjustele. [3]	18
Joonis 6. BCADS süsteemi arhitektuur. [7]	19
Joonis 7. Termokaamera tööprotsess. [8]	21
Joonis 8. Objekti tuvastus. [13]	22
Joonis 9. R-CNN arhitektuur. [12]	23
Joonis 10. YOLO arhitektuur. [12].....	23
Joonis 11. Semantiline segmenteerimine. [13].....	24
Joonis 12. Isendi segmenteerimine. [13]	24
Joonis 13. Mask R-CNN arhitektuur. [12]	25
Joonis 14. Panoptiline segmenteerimine. [18].....	25
Joonis 15. Töö valmimise protsess	28
Joonis 16. Termokaamera videofailist eksporditud pilt.....	29
Joonis 17. COCO JSON formaadi näide	31
Joonis 18. Käesoleva probleemi jaoks rakendatud siirdeõppe skeem	32
Joonis 19. Treenimata YOLOv5 mudeli rakendamine loomade termoandmestikul	36
Joonis 20. Treenimata Mask R-CNN mudeli rakendamine loomade termoandmestikul	37
Joonis 21. Näide treeningandmekogu pildist.....	38
Joonis 22. Näide annoteeritud treeningandme-	38
Joonis 23. Mask R-CNN mudeli treenimise kao graafik.....	40
Joonis 24. Treenitud mudeli rakendamisel saadud tulemus	40
Joonis 25. Treenitud Mask R-CNN mudeli kaofunktsiooni kõver	41
Joonis 26. Mask R-CNN mudeli rakendamisel saadud õigesti lokaliseeritud ning kategoriseeritud objektid	42
Joonis 27. Mask R-CNN mudeli rakendamisel saadud õigesti lokaliseeritud, kuid valesti kategoriseeritud objekt	42
Joonis 28. Treenitud Fast R-CNN mudeli kaofunktsiooni kõver	43
Joonis 29. Fast R-CNN treenitud mudeli õigesti lokaliseeritud ja kategoriseeritud tulemus	43

Joonis 30. Fast R-CNN treenitud mudeli õigesti lokaliseeritud, kuid valesti kategoriseeritud tulemus.....	43
Joonis 31. YOLO mudeli treenimisel saadud mAP.....	44
Joonis 32. YOLO mudeli õnnestunud ennustus	45
Joonis 33. YOLO mudeli ebaõnnestunud ennustus.....	45
Joonis 34. Termokaamera pilt kitsedega päikesepaistelises tingimustes	48
Joonis 35. Termokaamera pilt vihmastes tingimustes.....	48
Joonis 36. Termokaamera pilt lumistes tingimustes.....	49

Tabelid

Tabel 1. Ühe kategooriaga annoteeritud testandmestiku treenimiseks kohandatud hüperparameetrid	39
Tabel 2. Mitme kategooriaga annoteeritud testandmestiku treenimiseks kohandatud hüperparameetrid	41
Tabel 3. Fast R-CNN mudeli treenimiseks kohandatud hüperparameetrid.....	43
Tabel 4. Fast R-CNN mudeli treenimise tulemuste hindamine kategooriate järgi.....	44
Tabel 5. Närvivõrkude treenimise tulemused.....	46
Tabel 6. Väikeste ja keskmiste objektide tuvastustäpsus	50

1 Sissejuhatus

Igal aastal registreeritakse Eesti teedel keskmiselt 3000 sõiduki ja looma kokkupõrget, mille tagajärjeks on metslooma hukkumine ning kahju inimestele või nende varale. Üheks loomade hukkumise põhjuseks on suur liiklustihedus. Tiheda liiklusega maanteed killustavad metsloomade elupaiku. Viimastel aastatel on probleem pigem suurenenud, sest sõidukeid liigub Eesti teedel aina rohkem ja sõraliste arvukus on tänu pehmetele talvedele tõusnud. [1]

Transpordiamet koostöös keskkonnakonsultatsioonifirmaga Hendrikson&Ko analüüsis ja positsioneeris aastatel 2014–2018 riigimaanteedel toimunud loomaõnnetused ning koondas need registrisse. Ligi 44% registrisse kantud õnnetustest toimus põhimaanteedel, millest enim õnnetusi juhtus Tallinna–Tartu–Võru–Luhamaa maanteel. Eesti riigimaanteedel tuvastati statistiliselt olulisi loomaõnnetuste koondumiskohti kokku 322,2 km ulatuses, mis on võrdlemisi suur maa-ala ning on vajalik muuta nii liiklejatele kui ka loomadele ohutumaks. Kui loomi varakult tee-äärsetelt aladelt tuvastada, siis on võimalik juhtidele ohust teada anda ning vältida ulukiõnnetusi. [2]

Töös kasutatakse metsloomade tuvastamiseks termopilte ning konvolutsioonilisi närvivõrke. Erinevalt laialt kasutatava värvikaamera (RGB) piltidest on termokaameratega võimalik näha objekte ka piiratud nähtavusega tingimustes. Objekti tuvastuseks on varasemalt välja töötatud mitmeid konvolutsioonilisi närvivõrkude mudeleid, mis on treenitud suurte andmekogude peal. Selliste mudelite otsene rakendamine termopiltidel ei anna rahuldavaid tulemusi, kuna mudelid on treenitud RGB piltidel ning mainitud andmestikus ei ole vajalikke loomaliikide kategooriaid. Loomade tuvastamiseks termopildilt on vajalik närvivõrgu mudelid üle treenida probleemispetsiifiliste andmetega.

Käesoleva töö eesmärk on töötada välja konvolutsioonilise närvivõrgu mudel, mis suudaks tuvastada metslooma objekti termopildilt. Töö käigus on plaan luua loomade andmekogu, mis koosneks metsloomadest jäädvustatud termopiltidest, mis on salvestatud maanteeäärsetel aladel. Analüüsi käigus otsitakse vastuseid järgmistele uurimisküsimustele:

- Kuidas ning milliseid tehisnägemise (*computer vision*) meetodeid kasutades saab tuvastada metsloomi termopiltidelt?

- Kuidas mõjutavad erinevad ilmastikutingimused termopiltidelt objekti tuvastust?
- Kui täpselt suudab mudel objekti tuvastada erinevate kauguste pealt? Kas tuvastamise täpsus langeb kaugemate objektide puhul?

Uurimustöö on läbi viidud Eestis kasutusel oleva loomatuvaustussüsteemi parendamise perspektiivist, mida haldab Transpordiamet. Transpordiamet koostöös ettevõttega AS Kiirwarren on originaalandmete omanikud. Originaalandmed ei sisalda isikuandmeid ega muud tundlikku teavet, neid kasutatakse ainult käesoleva projekti käigus tehtava teadustöö jaoks.

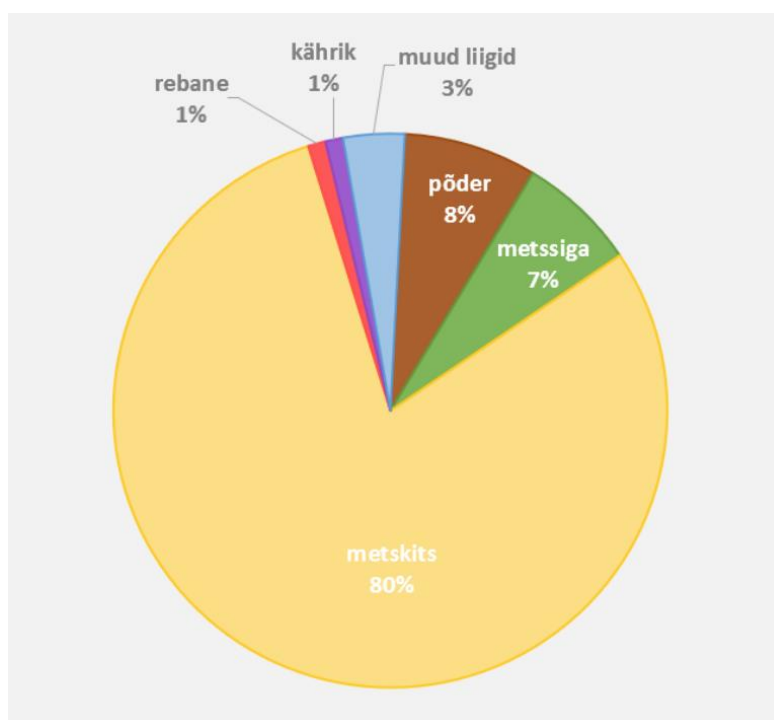
Kogu tehniline lahendus koos koodiga on leitav Github repositooriumist¹.

¹ <https://github.com/triinruutli/wild-animal-detection>

2 Ulukiõnnetused ning loomatu vastussüsteemid

2.1 Ulukiõnnetuste statistika

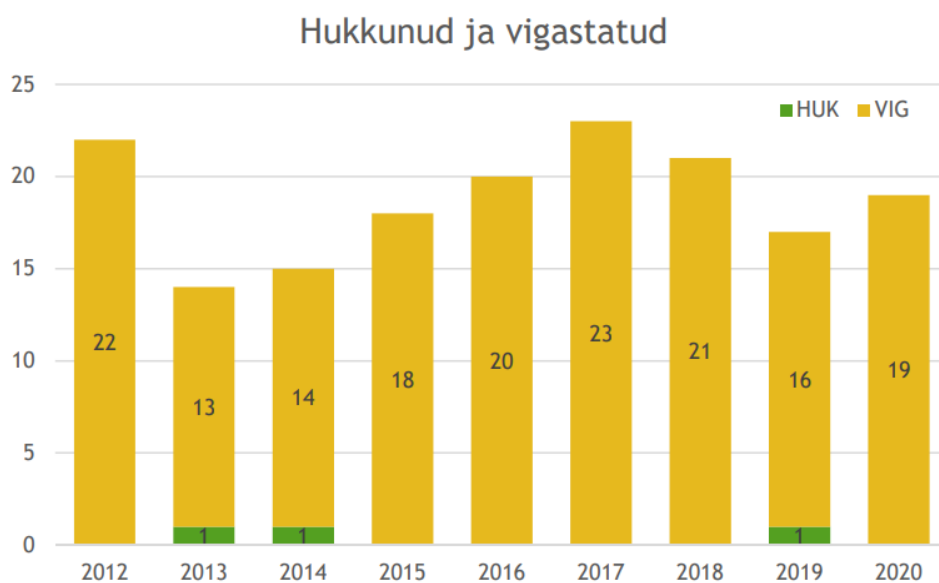
Viie aasta jooksul (2014–2018) registreeriti 18 401 teadet ulukiõnnetuste kohta, millest 11 059 olid piisavalt informatiivsed ning täpsed, et neist loodi Eesti ulukiõnnetuste andmebaas. Andmebaasist nähtub, et põhiosa registrisse kantud õnnetustest (ligi 95%) on juhtunud suurulukitega ning neist valdav osa metskitsedega (ligi 80%). Põtradega õnnetusi registreeriti andmebaasis 856 juhul ehk ligi 8%, ja metssigadega seotud õnnetusi vastavalt 767 ehk 7%. Suurkiskjate õnnetusi toimub maanteedel kordades vähem: hundi, karu ja ilvese õnnetusi võimaldas teate täpsusaste kanda andmebaasi kokku 18 korral. Väikeulukitest kanti registrisse kõige rohkem kähriku ja rebase õnnetusi, vastavalt 110 ja 113 korda. Enamik väikeulukitega seotud liiklusõnnetustest jääb registreerimata, seega siinkohal ei ole statistika reaalsusega kooskõlas. Lisaks nimetatutele kanti andmebaasi veel järgmiste liikidega toimunud õnnetusi: hirm, mäger, jännes, kobras, saarmas, siil, nugis, nirk. Ulukite jaotumine on välja toodud joonisel 1.[2]



Joonis 1. Andmebaasi kantud ulukite liigiline koosseis. [2]

Vigastatud või hukkunud inimestega metsloomaõnnetusi on olnud vähe. Aastatel 2021-2020 on hukkunutega liiklusõnnetusi olnud kokku 3, vigastatute arv on see-eest kordades

suurem, jäädes aastas keskmiselt 18 inimese juurde. Detailsem jaotus on nähtav jooniselt 2.



Joonis 2. Hukkunute ja vigastatute arv kokkupõrgetes loomadega. [3]

Registreeritud ulukiõnnetustest on sama uurimustöö raames koostatud ka looduslike ohutegurite analüüs, mis aitab leida seoseid ning põhjuseid suurele õnnetuste arvule. Looduslike ohutegurite analüüsi eesmärk oli analüüsida Eesti riigiteedel registreeritud ulukiõnnetuste seoseid maastikutunnustega. Kõrgem ulukiõnnetuste oht on metsasid läbivatel teelõikudel, metsa servades ja puisturibade läheduses ning madalam oht on lagedatel aladel ja hoonete läheduses. [2]

2.2 Metsloomade käitumismustrid ning nende seos õnnetustega

Ulukiõnnetusi on võimalik vähendada, kui me teame loomade käitumismustreid ning oskame nendega arvestada ka liikluses.

Üheks oluliseks teguriks on siinkohal loomade rändeajad, mis on loomade jaoks kõige aktiivsem liikumise periood. Kuna enamik ulukiõnnetustest on seotud sõralistega, siis keskendun ainult sõralistega seotud olulistele perioodidele. Aktiivsust suurendab jooksuaeg - metskitsedel juulis-augustis, põdral ja punahirvel septembris-oktoobris, metssigadel aasta lõpus. Põdralehmade poegimisaeg jääb maikuisse ning veel kogenematud eelmise aasta vasikad jäävad omapäi ja võivad maailma avastades sattuda ka maanteede äärde liiklust uudistama. [4]

Samuti on oluline roll loomade liikumisel ilmastikul. Pehmetel talvapäevadel vihma või uduga liiguvad sõralised rohkem kui selge taeva ning pakasega. Tugeva tuule korral hakkab loom otsima paremat varjet ja vältima kõrget metsa, mis paneb ta samuti liikvele. Suvel-sügisel uduste ja vihmaste ilmadega võivad metskitsed, aga ka põdrad olla aktiivsed just valgel ajal, selge ilma korral pigem hämarikus ja pimedas. Sügisene pimedaja pikenemine koos sademetega halvendab liiklusolusid sedavõrd, et ulukile otsasõidu risk suureneb kordades. [4]

Sarnaselt jooksuaajale paneb loomad aktiivsemalt liikuma ka jahihooaeg. Ühisjahtide aeg toimub oktoobrist jaanuari-veebruari, kui ulukid aetakse ajujahtides liikvele ja nad võivad sagedamini teid ületada, samas jätkub jooksuaeg nii hirvedel, põtradel kui ka metssigadel. Jaht toimub tavaliselt puhkepäevadel, kui liiklust on rohkem, mistõttu peavad autojuhid olema eriti valvsad. [4]

2.3 Loomatuvastussüsteemid

Maanteedel on kasutusele võetud mitmeid ennetavaid meetmeid, et vältida metsloomadega seotud õnnetusi ning tagada neile turvaline läbipääs. Siinkohal võetakse fookusesse ainult tehniliselt automatiseeritud lahendused.

Automatiseeritud loomatuvastussüsteemil (LTS) on positiivne mõju liiklusohutusele järgmistel eeldustel:

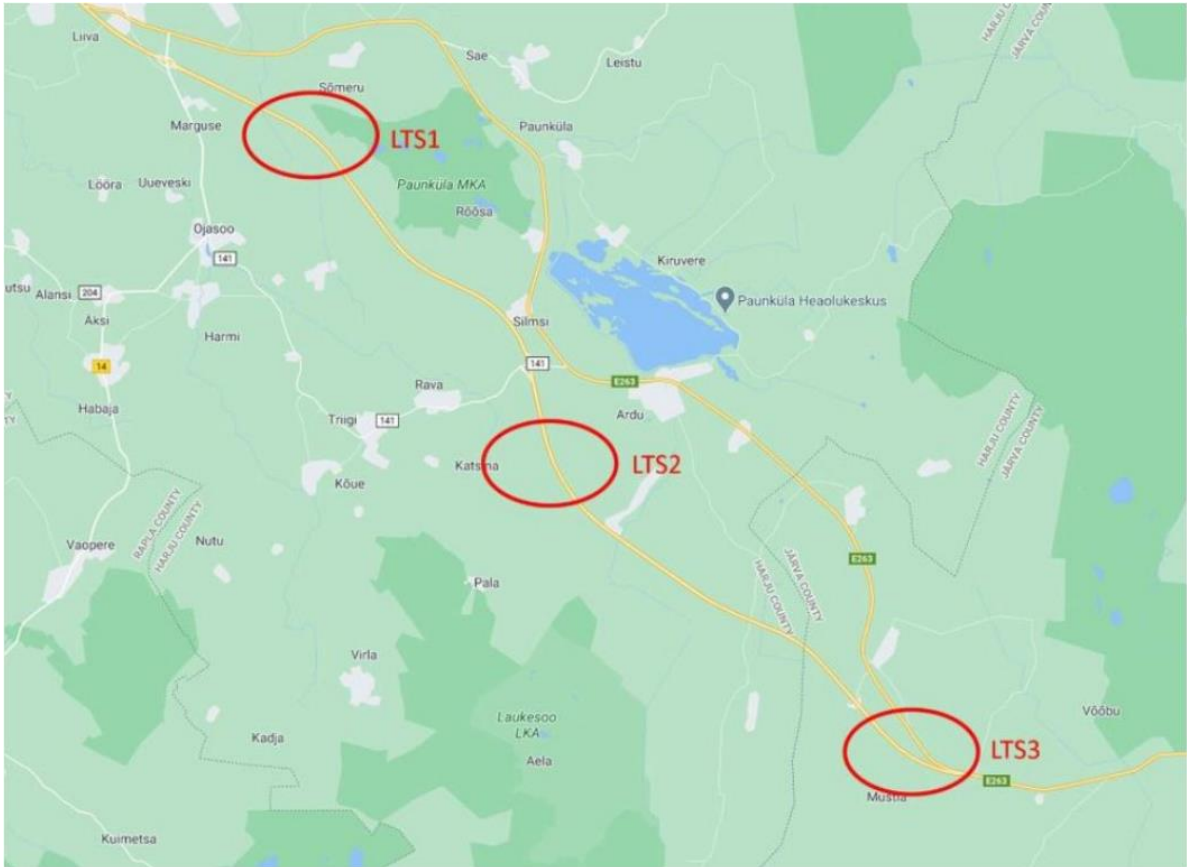
- Reaalajas ja loomade seirel tuginev süsteem toimib liiklejate arvates täpsemini ja usaldusväärsemalt, kui lihtsalt hoiatusmärkidel põhinev lahendus. [3]
- Juhid järgivad võimalikku rakendatud adaptiivset piirangut paremini kui mitteadaptiivset (tavalise liiklusemärgiga edastatud hoiatus või kiiruspiirang). [3]

Kui mainitud eeldused on täidetud, siis tooks süsteemi kasutamine kaasa autode sõidukiiruse olulise languse, mis omakorda suurendab ohutust, vähendades kokkupõrkeriski. Seega juhul on ohule reageerimise reaktsiooniaeg suurem ning väiksema kokkupõrkekiiruse korral on ka õnnetuse tagajärjed kergemad. [3]

2.3.1 Eestis kasutusel olev loomatuvastussüsteem

Eestis on loomatuvastussüsteem kasutusele võetud ainsana Tallinn-Tartu maanteel Kose-Võõbu teelõigul, kus on kolm metsloomade teeületusala koos automaatse

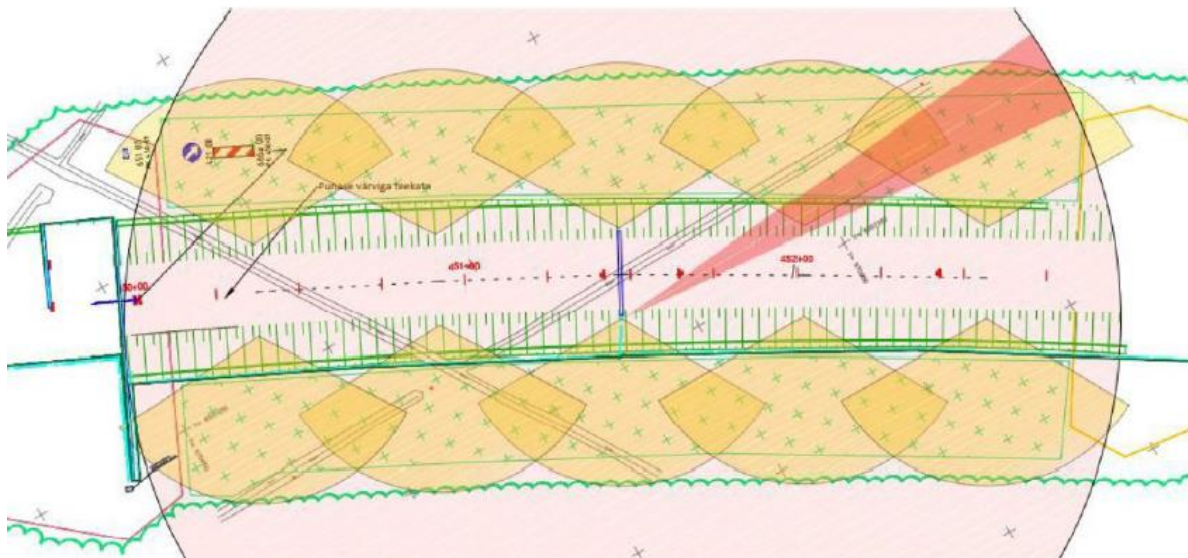
teavitussüsteemiga muutteabe märkidel. Tuvastussüsteemide asukohad on järgmised: LTS1 maantee nr 2 45,1 kilomeetril, LTS2 53,5 kilomeetril, LTS3 63,4 kilomeetril (joonis 3). [3]



Joonis 3. Loomatuvastussüsteemide asukohad Eestis. [3]

Jälgimissüsteemi põhiosa moodustavad 10 lähijälgimisradarit, mis skaneerivad maanteeäärset ala ja liikuva objekti ilmumisel radari vaatevälja edastavad juhtkontrollerile avastatud objekti koordinaadid, samal ajal jätkates avastatud objekti liikumistrajektoori salvestamist. Juhtkontroller edastab avastatud objekti koordinaadid pöördtermokaamera ohjamistarkvarale, mis suunab kaamera avastatud objektile ja radaritelt saabuva teabe alusel hoiab avastatud objekti pidevalt kaadris, kuni selle väljumiseni jälgimisalast (joonis 4). Samal ajal edastatakse informatsioon muutuvsisuga liiklusemärgidele. Kummalgi pool teed on 300 meetrit enne teeületusala algust ja 100 meetrit enne teeületusala lõppu paigaldatud muutuvteabega liiklusemärgid (VMS), mis kuvavad vastava info, kui valvealal fikseeritakse intsident. Tuvastussüsteem ja VMS märgid on kaughallatavad Transpordiameti

liiklusjuhtimiskeskusest. Kõik intsidendid salvestatakse ja logid on järeltöödeldavad süsteemi töö seirel. [3]



Joonis 4. Radarite ning kaamerate paigutus. [3]

Siinkohal on oluline veelkord välja tuua, et tuvastamine toimub ainult radarite abil ning termokaameratele antakse tuvastuse korral signaal, mis käivitab videofaili. Termokaamera salvestusi kasutatakse vaid teed ületavate loomade statistika loomiseks. Pärast viimase liikumise tuvastamist algab kontrollperiood, mille jooksul kaamera salvestab veel 2 minutit.[5]

Loomatuvastussüsteemis kasutatakse Axis Q1941-E PT¹ pöördmehhanismiga termokaamerat. Kaamerapilt katab kogu teeületuspiirkonna, mis on kokku 3,6 ha. Lähimas kohas on metsapiir kaamerast umbes 40 meetri kaugusel, kaugeimas kohas aga kaugelt üle 150 meetri. Termokaamera on asetatud teeületuskoha keskele, olles mõlemast küljest võrdsel kaugusel ületuskoha algusest.[5]

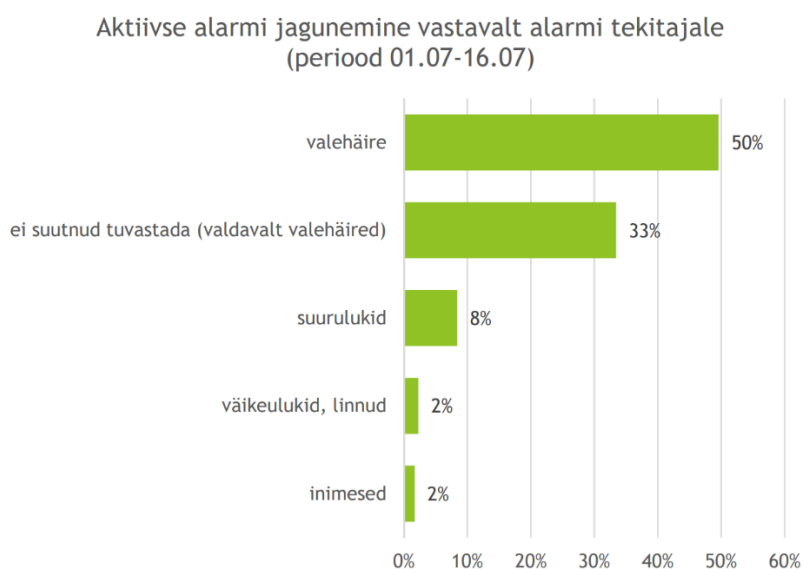
See lahendus on töötanud alates 2020. aasta augustikuust. Süsteemi teostatavusuuringu käigus analüüsiti 2 kuu andmeid ning kogutud andmete põhjal on näha, et esineb palju valepositiivseid tuvastusi ehk olukordi, kus radar tuvastab liikumise, aga termokaamera looma ei leia. Selliste tuvastuse osakaal on ligi 80%, sealhulgas 50% on tuvastatud põhjusega valehäired ja 30% on häire põhjus jäänud tuvastamata. Levinumate valehäirete

¹ <https://www.axis.com/products/axis-q1941-e-pt-mount>

põhjuseks tuuakse välja tuul ehk tuule mõjul liikuvad objektid, vihm ja radarite interferents. Samas nende esinemise osakaal ei ole teada. [3]

2.3.1.1 Olemasoleva süsteemi parendamine

Valepositiivsete tuvastuste osakaal on väga suur – umbkaudu 80% (joonis 5) [3]. Selliste tuvastuste põhjused on vaja välja selgitada, et süsteemi oleks võimalik paremaks muuta.



Joonis 5. Loomatuvastussüsteemi alarmi jagunemine vastavalt põhjustele. [3]

Valepositiivsete tuvastuste suur osakaal võib viidata sellele, et kuna radareid on ühes vaatlusalas kokku 10, siis need radarid võivad reageerida tagasipeegeldustele või segavad üksteise tööd. Radarite arvu vähendamisega on võimalik selliste valepositiivsete tuvastuste arvu vähendada, kuid alternatiivsed lahendused, nagu fikseeritud Doppleri radar [6] või 360 kraadi skaneeriv radar on võrreldes fikseeritud radaritega kordades kallimad. 360 kraadi skaneeriva ühe radariga süsteemi maksumus on orienteeruvalt 100 000 - 120 000 eurot. [3]

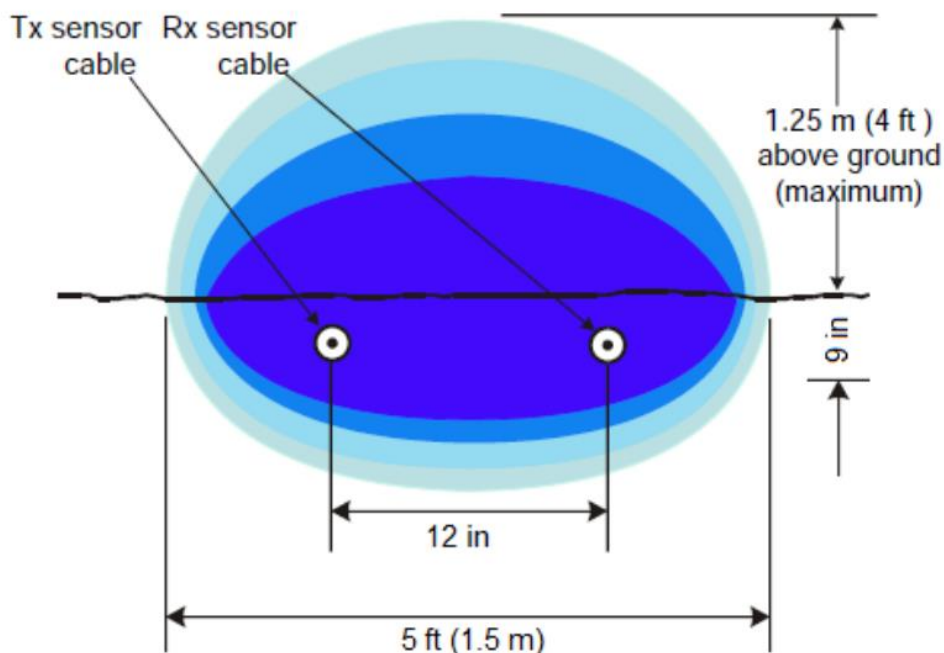
Samuti on üks võimalus valepositiivsete tuvastuste vähendamiseks võtta lisaks radari anduritele kasutusele mõni teist tüüpi andur ja anda informatsioon muutuva teabega märkidele ainult siis, kui mõlemad andurid on kinnitanud looma tuvastamise. Teiseks anduriks võiks olla termokaamera, mis peab looma enne info edastamist tuvastama. Siinkohal oleks väga oluline käesoleva töö käigus tehtud analüüs, saadud tulemust saaks rakendada koostöös radaritega või piisava töökindluse korral ka radaritest eraldiseisva süsteemina.

2.3.2 Loomatuvastussüsteemide erinevad tehnoloogiad

Maailmas kasutatavad loomatuvastussüsteemid põhinevad erinevatel anduritel ning tuvastamise viisidel. Käesolevas alampeatükis tutvustatakse üht alternatiivset lahendust, mis on edukalt kasutusele võetud Ameerika Ühendriikides ning tutvustatakse üldiselt süvaõppe lahendustel põhinevaid süsteeme.

2.3.2.1 Pinnases asuva kaabliga loomatuvastussüsteem

Pinnases asuva kaabliga loomatuvastussüsteem (*Buried cable animal detection system (BCADS)*) on joontuvastussüsteem, milles kaks kaablit on paigutatud üksteisest 30 cm kaugusele 25 cm sügavusele maa sisse. Üks kaabel on edastuskaabel (TX), mis jaotab raadiosageduslikud signaalid mööda kaabliteed ja teine on vastuvõtukaabel (RX), mis võtab signaalid vastu ja tagastab need protsessorile (joonis 6). Raadiosageduslikud signaalid moodustavad kaablite ümber nähtamatu elektromagnetilise tuvastusvälja, mis suudab tuvastada seda välja läbiva looma ja määrata selle asukoha. Seejärel käivitab protsessor loomade kohaloleku põhjal häire. [7]



Joonis 6. BCADS süsteemi arhitektuur. [7]

Virginias tehtud uuringu kohaselt suudab kõnealune süsteem tuvastada suurulukeid nagu põder > 98% täpsusega. Väikeulukeid, nt mäger või rebane suudab süsteem tuvastada õige konfiguratsiooni ja kalibreerimise korral. [7]

BCADS ei ole sõltuv ilmastikutingimustest. Süsteem suudab töötada ka 60 cm paksuse lumekihi all. Paksema lumekihi korral võib tekkida valenegatiivseid olukordi, mis tähendab, et süsteem ei suuda looma tuvastada. Samuti on antud süsteem sõltumatu tuulepuhangutest ning vihasadudest. Süsteemi peamine murekoht on, et tegemist ei ole ala katva süsteemiga, vaid joontuvastussüsteemiga. Seetõttu ei teki peale joonte ületamist ülevaadet, kas loom on alast lahkunud või on endiselt tuvastusalas. See omakorda hakkab mõjutama hoiatusmärkidele kuvatavat infot ning ajab autojuhid segadusse. [7]

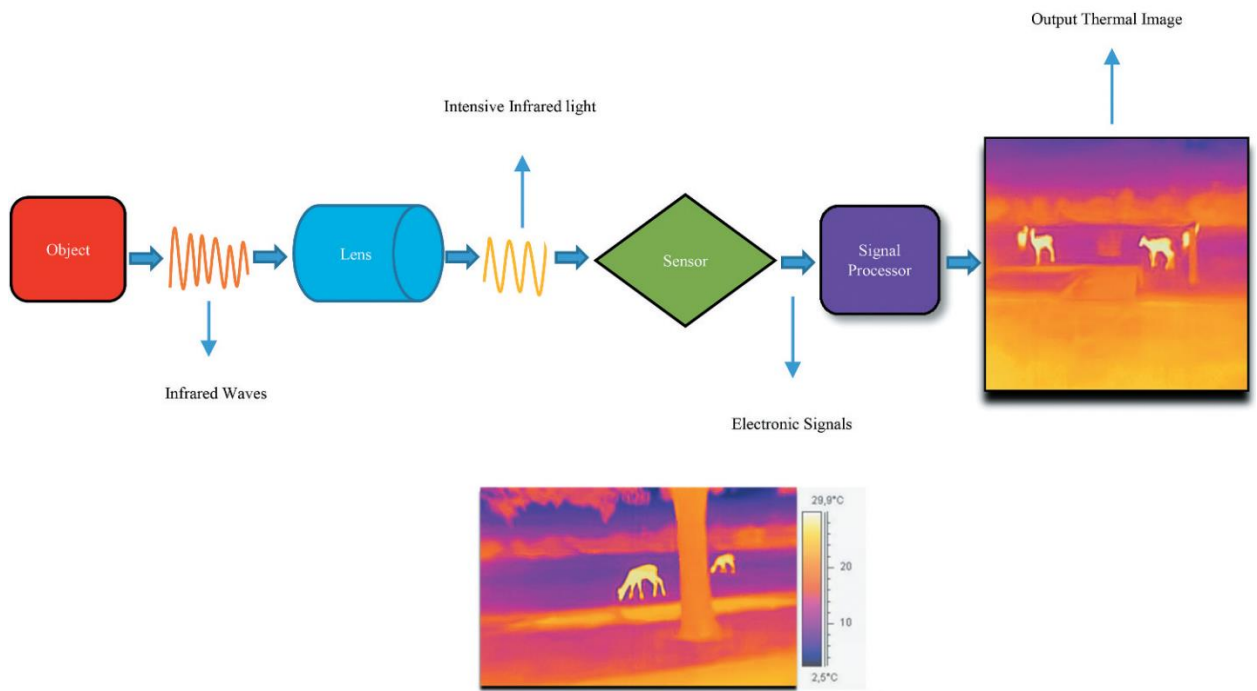
2.3.2.2 Süvaõppel põhinevad süsteemid

Pilditöötuse ning süvaõppe meetodite abil on võimalik tuvastada kaamerapildist edukalt objekte. Süvaõppe mudelid on pilditöötusrakendustes kõige tõhusamad, kuna need eristavad pildi kõrgetasemelisi omadusvektoreid madala taseme omadusvektoritest.

Loomade liikumine toimub enamjaolt hämaras või pimedas, seega tavalise RGB kaameraga ei ole võimalik eristada nähtavat objekti. Seetõttu kasutatakse mitmel pool maailmas termokaameraid, mis võtab vastu objektilt soojuse emissiooni ja genereerib liitpildi, mis kuvab objekti ligikaudse temperatuuri ning muudab objekti vähese valguse tingimustes nähtavaks. [8]

Termopildid väljendavad mõõdetud objektist kiirguvat soojust infrapunakiirguse kujul. Kõige kuumemad piirkonnad tekitavad tugevat kiirgust ja see omakorda suurendab pildi intensiivsust või selgust. Termokaamera jäädvustab infrapunakiirguse pikema lainepikkuse, võimaldades objekti täpselt tuvastada. Termopildil on nii jahedamad kui ka soojemad objektid. [8]

Infrapunakiirguse püüdmiseks on vaja kaameraobjektiivide komplekti, mis osutab objektile ja keskendub kõigi ümbritsevate objektide kiirgavale infrapunavalgusele. Detektorite hulk skaneerib fokuseeritud valgust ja luuakse täpne temperatuurimuster, mida nimetatakse termogrammi kujutiseks. Seejärel tõlgitakse termogramm impulssideks, mis saadetakse signaaliprotsessorile ning seejärel muudab kaamera sees olev kiip impulsid kuvapiltideks. Kogu protsess on välja toodud alloleval joonisel 7. [8]



Joonis 7. Termokaamera tööprotsess. [8]

Termokaameratelt loomade tuvastamiseks kasutatakse erinevaid süvaõppemeetodeid. Enamjaolt rakendatakse kaamerapildil konvolutsioonilisi närvivõrke ning tuvastamise korral käivitatakse tegevus või alarm, mis hoiatab liiklejaid.

Toetudes varasematele teadusartiklitele, kus võrreldakse mitme konvolutsioonilise närvivõrgu rakendamise tulemusi termokaameratel, on piirdekasti meetodil (*bounding box*) vaieldamatult parimaid tulemusi nii täpsuse kui kiiruse osas näidanud YOLO arhitektuuriga närvivõrk [9]. Lisaks YOLO arhitektuurile on lootustandvaid tulemusi täpsuses näidanud ka Fast R-CNN mudel, kuid valideerimise kiirus on tunduvalt suurem YOLO mudelil [9]. Samuti on varasemalt uuritud, kuidas toimib semantiline segmenteerimine termoandmetel, selleks on rakendatud spetsiaalse arhitektuuriga närvivõrke nagu Mask R-CNN ning EC-CNN [10].

2.4 Tehisnägemise alamvaldkonnad

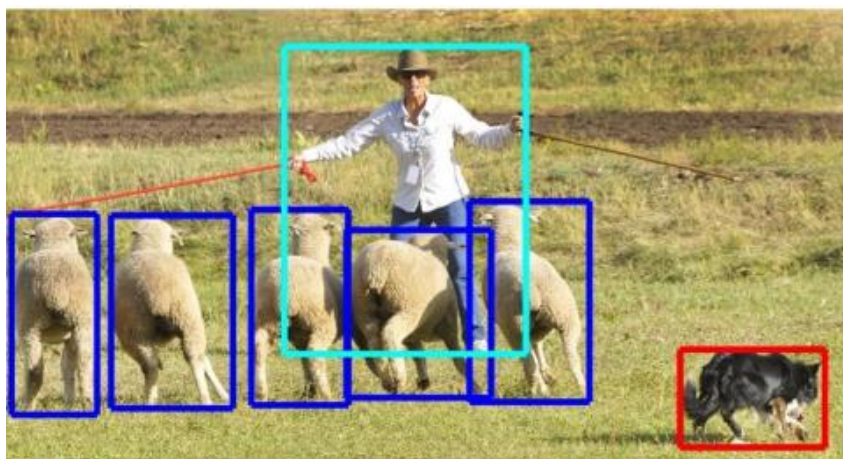
Arvutinägemine ehk tehisnägemine on tehnoloogia valdkond, mis põhineb tehisintellektil ning tegeleb pildilt informatsiooni leidmise ja töötlemisega. Selle valdkonna eesmärk on anda arvutile inimaju intelligentsus nii, et arvuti omaks piltidelt objektide äratundmis- ning analüüsisoskust. Olenevalt eesmärgist jaotatakse tehisnägemises esinevad tegevused

ülesanneteks (*computer vision tasks*) või alamklassideks. Levinumad ülesanded on objekti klassifitseerimine, objekti tuvastus, pildi segmenteerimine, näotuvastus.[11]

Kuna käesolevas töös keskendutakse objekti tuvastusele ning segmenteerimisele, siis käsitleb see peatükk ainult neid alamklasse.

2.4.1 Objektituvastus

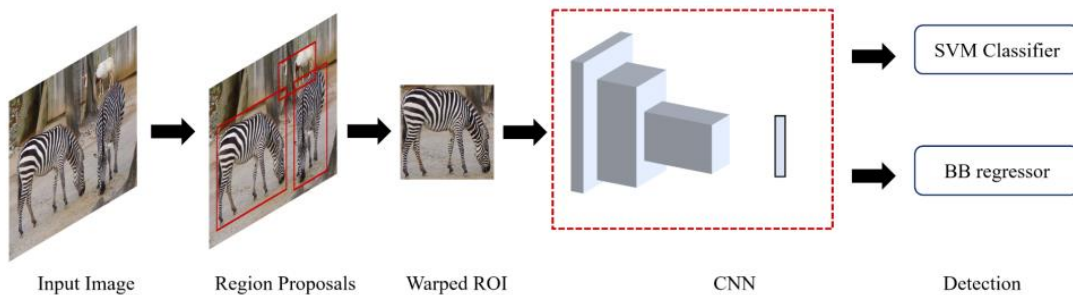
Objektituvastus on tehisenägemise valdkonna üks alamklass, mis keskendub objektide klassifitseerimisele ning lokaliseerimisele pildil või videol. Iga üksiku objekti asukoht määratakse pildil piiravate kastikestega (*bounding box*), näide joonisel 8. Objekti tuvastuseks kasutatakse peamiselt konvolutsioonilistel närvivõrkudel põhinevaid meetodeid. [12]



Joonis 8. Objekti tuvastus. [13]

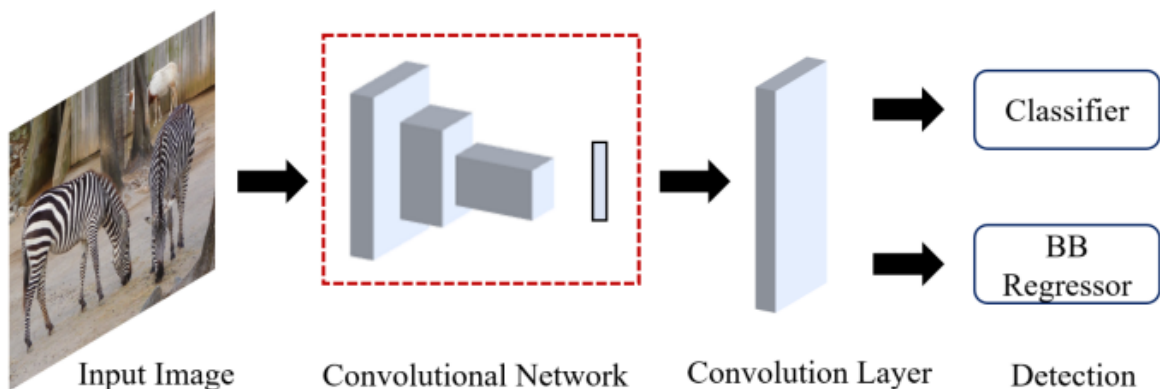
Objekti tuvastuse mudelid jagunevad põhimõtteliselt kaheks: üheastmelised ning kaheastmelised mudelid. Närvivõrku, millel on eraldi moodul piirkonna ennustuste loomiseks (*region proposal*), nimetatakse kaheastmeliseks detektoriks. Need mudelid püüavad esimeses etapis leida pildil suvalise arvu objektide ennustusi ning seejärel klassifitseerida ja lokaliseerida need teises etapis. Üheks selliseks näiteks on regioonipõhine konvolutsiooniline närvivõrk (*Region-based Convolutional Neural Network - R-CNN*). R-CNN mudelil on mitmeastmeline treenimisprotsess, kus esmalt eel-treenitakse närvivõrku mahuka andmekogu peal, millel on palju kategooriaid ning seejärel treenitakse mudel üle valdkonnaspetsiifiliste andmetega, mida kasutatakse käesoleva probleemi lahendamiseks. R-CNN arhitektuurist on edasi arendatud Fast R-CNN ning Faster R-CNN arhitektuurid. Kuna kaheastmelistel detektoritel on kaks eraldi

etappi ning nende arhitektuur on keerulisem, siis kulub nendel mudelitel ettepanekute loomiseks rohkem aega (joonis 9). R-CNN mudelid on näidanud tuvastamise täpsuses väga häid tulemusi. [12]



Joonis 9. R-CNN arhitektuur. [12]

Üheastmelised detektorid klassifitseerivad ja lokaliseerivad semantilised objektid ühe korraga, kasutades selleks erineva skaala ja kuvasuhtega eelmääratletud kastikesi. Üheastmelised detektorid on näiteks YOLO mudelid (*you-only-look-once*). YOLO mudelis jaotatakse sisendpilt $S \times S$ ruudustikuks ning lahter, millele langeb objekti keskpunkt, vastutab selle tuvastamise eest (joonis 10). Tuvastamise käigus ennustatakse mitu piiravat kasti (*bounding box*) ja iga ennustus koosneb järgmistest elementidest: piirdekasti keskpunkt, x ning y koordinaadid, piirdekasti mõõtmed w ja h ning usaldusskoor. Üheastmelised mudelid on lihtsama disainiga ning treenimiseks kulub oluliselt vähem aega kui kaheastmeliste mudelite puhul. Lisaks on treenitud mudeli rakendamine valideerimisandmetel väga kiire, seega tänu oma kiirusele on YOLO mudelid hästi rakendatavad videotel. [12]



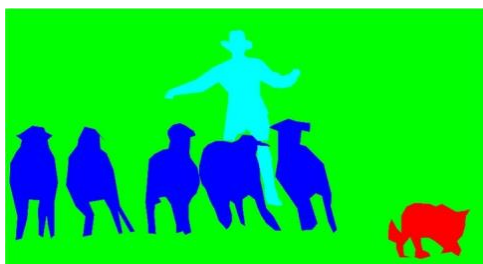
Joonis 10. YOLO arhitektuur. [12]

2.4.2 Pildi segmenteerimine

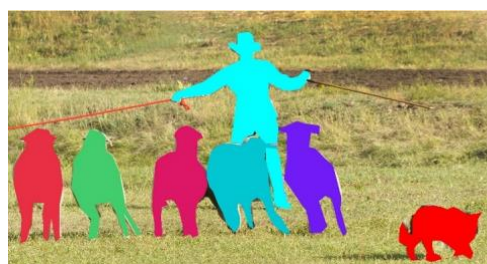
Pildi segmenteerimine on tehisnägemise oluline ülesanne (*computer vision task*), mille käigus jaotatakse pilt mitmeks segmendiks ning objektiks. Segmenteerimine jaguneb oma olemuselt kolmeks: semantiline segmenteerimine (*semantic segmentation*), isendi segmenteerimine (*instance segmentation*) ning panoptiline segmenteerimine (*panoptic segmentation*).

Semantilise segmenteerimise puhul omastatakse igale pildi pikslile väärtus objekti kategooriast [14]. Objektide kategooriad on erinevad klassid, näiteks inimene, koer, auto, puu jne. Lisaks arvutatakse tehtud kategoriseerimise abil välja objekti kujule vastav segmentatsiooni mask (*segmentation mask*) [14]. Erinevalt tuvastamise ülesandest ei ole semantilise segmenteerimise puhul vaja eristada sama kategooria erinevaid isendeid [13]. See võimaldab märgistada objekte, mille üksikuid eksemplare on raske määratleda, näiteks muru, taevast, tänavad või seinad. Joonisel 11 on näha semantilist segmenteerimist.

Isendi segmenteerimine on olemuselt väga sarnane semantilisele segmenteerimisele, kuid oluline erinevus on individuaalsete eksemplaride tuvastamises. Isendi segmenteerimine ühendab tuvastamist ning semantilist segmenteerimist [15]. Joonisel 12 on näha näidet isendi segmenteerimisest.



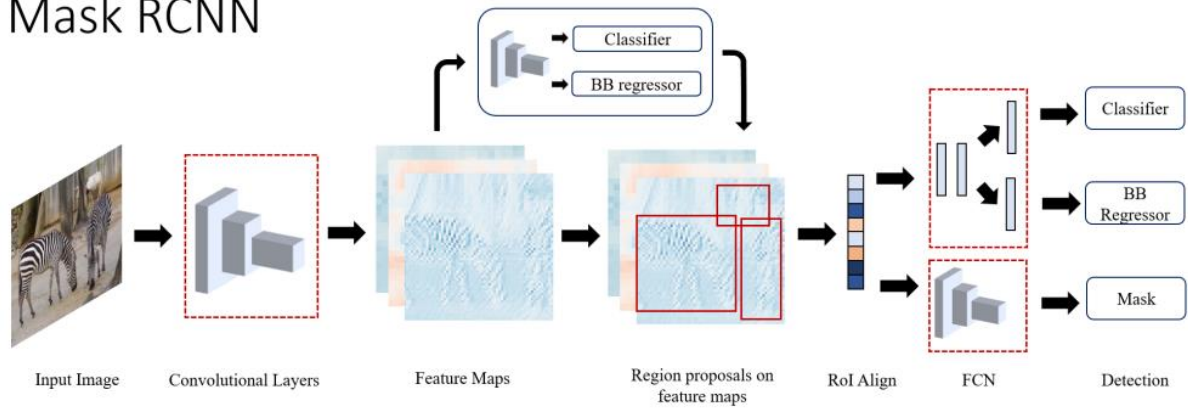
Joonis 11. Semantiline segmenteerimine. [13]



Joonis 12. Isendi segmenteerimine. [13]

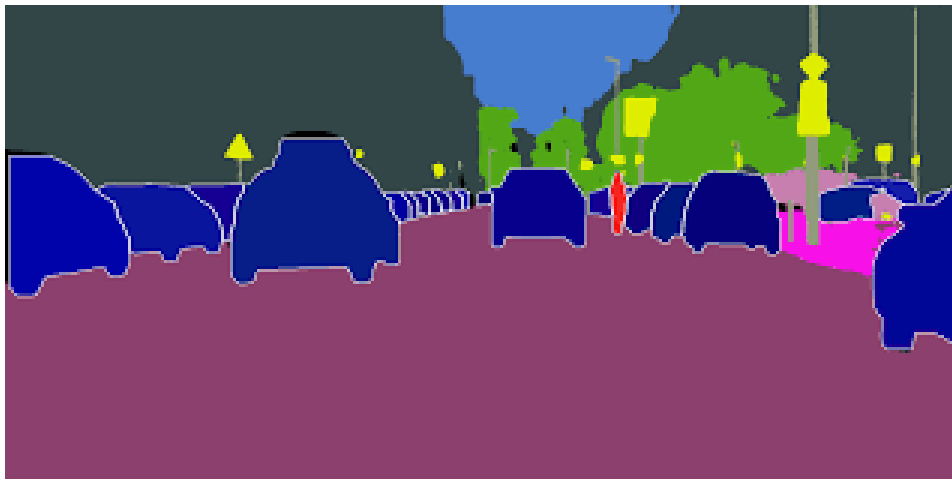
Mask R-CNN on Faster R-CNN arhitektuurist välja arenenud meetod isendi segmenteerimiseks, millel on lisaks klassifitseerimisele ja lokaliseerimisele paralleelselt lisakomponent segmenteerimismaskide ennustamiseks igas huvipiirkonnas (joonis 13). Erinevus Faster R-CNN arhitektuuriga tuleneb teises etapis, kus on tehtud oluline muudatus omaduste eraldamiseks kasutatava RoIPool kihi asendamises RoIAlign kihiga, mis parandab maski täpsust 10-15%, näidates häid tulemusi lokaliseerimismõõdikute korral. [16]

Mask RCNN



Joonis 13. Mask R-CNN arhitektuur. [12]

Panoptiline segmenteerimine on tehisnägemise ülesanne, kus pikslid kategoriseeritakse kahe klassi vahel: aine (*stuff*) ning asi (*thing*). Aine klassi kuuluvad amorfseid piirkonnad nagu muru, taevas, maantee ning asjade klassi kuuluvad loetavad objektid nagu inimene ja auto [17]. Mainitud kategooriad jagavad kõnealuse ülesande omakorda kaheks alamülesandeks, milleks on semantiline ja isendi segmenteerimine. Igale pildi pikslile määratakse kategooria silt ning unikaalne identifikaator, mis näitab, kas tegemist on objekti isendiga [18]. Sama sildiga pikslid kuuluvad samasse kategooriasse ning sama identifikaatoriga pikslid kuuluvad samale objekti isendile [18]. Joonisel 14 on näha panoptilise segmentatsiooni rakendamist.



Joonis 14. Panoptiline segmenteerimine. [18]

2.4.3 Mudeli tulemuste hindamine

Objektituvastuse ja segmenteerimise mudelite tulemusi valideeritakse peamiselt AP (*average precision*) mõõdiku alusel, mis tähistab keskmist täpsust. AP on tuletatud IoU

mõõdikust (*intersection over union*), mis tugineb *Jaccardi* indeksile, mis on kahe andmehulga sarnasuse koefitsent [19]. IoU on õigesti ennustatud pikslite ning pikslite koguarvu suhe iga kategooria kohta (*võrrand 1*). Kui IoU mõõdikule anda lävendiks väärtus t , siis saab hinnata, kas ennustus on õige ($\text{IOU} \geq t$) või ebaõige ($\text{IOU} \leq t$). [20]

$$J(B_p, B_{gt}) = \text{IOU} = \frac{\text{area}(B_p \cap B_{gt})}{\text{area}(B_p \cup B_{gt})}$$

Võrrand 1. IOU võrrand. [19]

Täpsus (*precision*) on mudeli võime tuvastada ainult asjakohane objekt kõigi tagastatud objektide seast. See on õigete positiivsete ennustuste protsent. Saagis (*recall*) on mudeli võime leida üles kõik märgitud põhitõe (*ground truth*) piirdekastid (*bounding box*) ning see on õigete positiivsete ennustuste protsent kõigi põhitõdede hulgas. Mõlemate mõõdikute väärtused on alati 0 ning 1 vahel. Ennustustel on usaldusskoorid (*confidence score*), mis järjestatakse kahanevas järjekorras ning mille alusel arvutatakse täpsuse ning saagise väärtused. Saadud väärtustest tekib joonisele kõver (*precision and recall curve*), mille abil arvutatakse AP mõõdiku väärtust. Suur kõveraalne pindala viitab suurele täpsusele ning suurele saagisele, mis omakorda tähendab, et mudelit võib hinnata heaks. Kuna tihti on mainitud kõver sik-sakiline ning põhjustab hindamisel probleeme, siis on vajalik kõverat interpoleerida. Selleks kasutatakse kõikide punktide interpoleerimist, mille käigus tähistatakse kõik unikaalsed saagise punktid (r_1, r_2, \dots), kui maksimaalne täpsuse (p) väärtus väheneb. \tilde{r} viitab saagise väärtusele leitud maksimaalse täpsuse juures. Võrrand 2 on AP mõõdiku arvutamiseks. [19]

$$AP = \sum (r_{n+1} - r_n) P_{interp}(r_{n+1}), \text{ kus } P_{interp}(r_{n+1}) = \max p(\tilde{r})$$

Võrrand 2. AP arvutamine. [19]

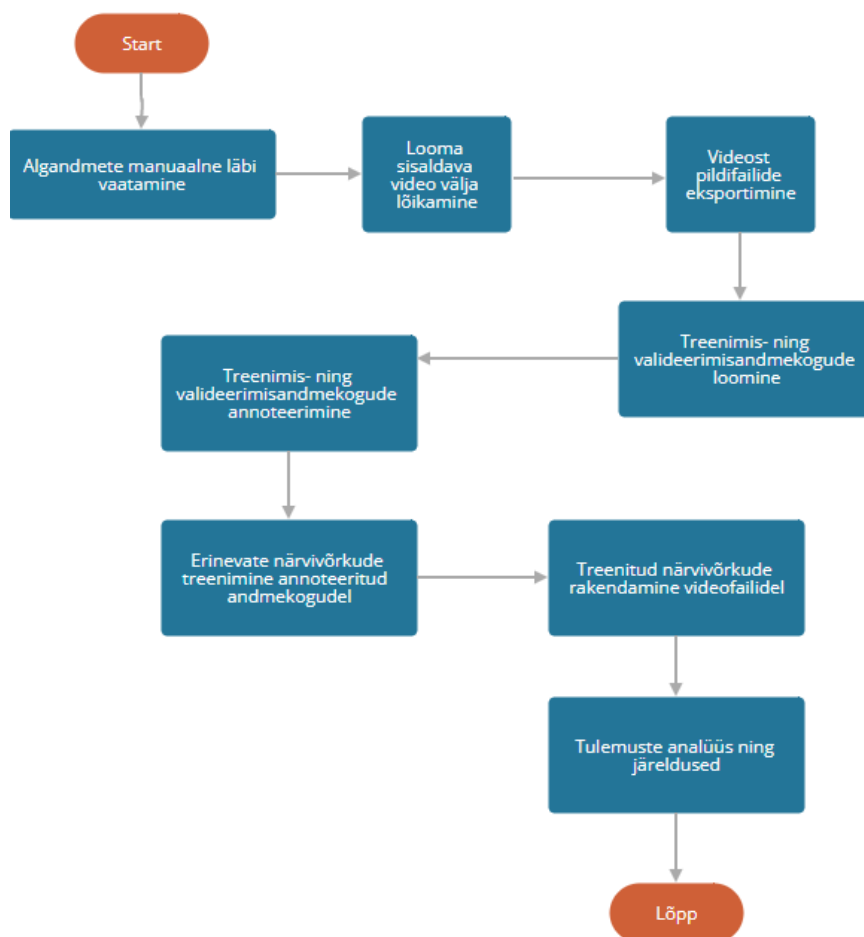
COCO andmestike puhul on AP arvutatud üle kõikide kategooriate ning IoU lävendite 0,5 kuni 0,95. See on kõige täpsem mõõdik, mida võiks COCO andmestikel treenitud mudelitel kasutada. Lisaks on erinevate suurustega objektide hindamiseks detailsemad mõõdikud nagu AP_{small} , AP_{medium} , AP_{large} . [21]

Panoptilise segmenteerimise jaoks kasutatakse PQ (*panoptic quality*) mõõdikut, mis on välja pakutud panoptilise segmentatsiooni autorite poolt. PQ on lihtne ning informatiivne mõõdik, mida saab kasutada kõikide ainete ning asjade kategooriate jõudluse mõõtmiseks ühtsel viisil. Teised olemasolevad mõõdikud suudavad hinnata kas asjade või ainete

kategooriaid eraldiseisvalt, kuid mitte koos, nagu on vajalik panoptilise segmenteerimise puhul. PQ leidmine koosneb kahest etapist: segmentide leidmine ning PQ arvutamine. Segmentide leidmisel tehakse kindlaks, et IoU oleks rangelt üle 0,5, sest see nõue aitab kindlustada, et tegemist on unikaalse segmendiga ehk leitud saab olla ainult üks ennustatud segment iga põhitõe segmendi kohta. PQ arvutamisel leitakse esmalt iga kategooria kohta PQ ning seejärel leitakse kõikide kategooriate keskmine. Nii on PQ väärtus sõltumatu kategooriate vahelisest tasakaalutusest. [17]

3 Metoodika ja realisatsioon

Käesoleva töö valmimise protsess koosnes mitmest olulisest etapist: andmete töötlemine, andmete annoteerimine, närvivõrkude treenimine, tulemuste valideerimine, analüüs ning järeldused (joonis 15). Selles peatükis kirjeldatakse mainitud etappe detailsemalt.



Joonis 15. Töö valmimise protsess

Esmalt keskendus autor sarnaste probleemide uurimisele teaduskirjanduses, et mõista, millised on siiani parimad lahendused töö probleemile. Sealhulgas otsiti ka olemasolevaid avalikult kättesaadavaid termopiltide andmestikke, mida võiks probleemi lahendamiseks kasutada. Üheks potentsiaalseks andmekoguks oli Teledyne FLIR ADAS andmestik, mis koosnes nii termopiltidest kui ka RGB piltidest [22]. Üpris kiiresti selgus, et käesoleva probleemi lahendamiseks on vajalik luua spetsiaalne andmekogu, sest avalikult kättesaadavaid termopiltide andmestikke on väga vähe ning nendes sisalduvad kategooriad ei ole relevantset.

3.1 Andmete kogumine ning puhastamine, andmekogu loomine

Töö kõige mahukamaks osaks on andmete kogumine ning töötlemine. Andmete kogumise faasis kontakteerus autor Eesti Transpordiametiga, kelle haldusalas on Kose-Võõbu maanteelõigul asuv loomatuvastussüsteem, mis salvestab termokaameraga loomade liikumist. Transpordiamet koostöös ettevõttega AS Kiirwarren on originaalandmete omanikud. Originaalandmed ei sisalda isikuandmeid ega muud tundlikku teavet, neid kasutatakse ainult käesoleva projekti käigus tehtava teadustöö jaoks. Kuna videofailid on väga mahukad, siis on need kättesaadavad ainult portatiivsel andmekandjal.

Algandmestik koosneb videofailidest, mis on jäädvustatud 2021. aasta maikuu kahel esimesel nädalal ning 2022. aasta märtsikuu jooksul. Erinevate ajavahemike kasutamise põhjuseks on võimalikult mitmekesiste ilmastikutingimustega videomaterjali saamine ning töötlemine. Maikuu failidel on ilmad peamiselt soojad ning päikselised, samas esineb ka vihma ning udu. Märtsikuu videofailidel on näha lund ning lumesadu.

Videofailide avamiseks ning töötlemiseks kasutati XProtect Smart Client tarkvara¹. Videofailid töötati manuaalselt läbi ning olulised videoklipid koos nähtavate loomadega on välja lõigatud. XProtect Smart Client võimaldab videoklippe ekspordida ka pildifailidena, näide on toodud joonisel 16. Eksporditud pildifailid on JPG formaadis ning mõõtmetega 480x360.



Joonis 16. Termokaamera videofailist eksporditud pilt

¹ https://doc.milestonesys.com/latest/en-US/standard_features/sf_sc/sf_reducedmanual/sc_clientexplained.htm

3.2 Andmete annoteerimine

Andmete annoteerimiseks kasutati mitmeid avalikult kättesaadavaid tööriistu. Esimene valik oli kasutada tuvastuseks piirdekastide (*bounding box*) meetodit, mis tekitab objekti ümber väikese kastikese. Selleks kasutati *makesense.ai* tööriista [23], mis võimaldab luua erinevate objektide jaoks silte ning nende siltide alusel piirdekastikestega objekt tähistada. Tekkinud annotatsioonid on eksporditavad üldkasutatavates formaatides nagu YOLO või VOC XML.

Kuna piirdekasti meetod ei suuda tuvastada ega edasi anda objekti kuju ning seetõttu pole kõige täpsem annoteerimise meetod, siis prooviti ka hulknurga (*polygon*) annoteerimist. Hulknurga annoteerimise puhul ühendatakse hulknurga tipud, mis on tähistatud ümber objekti nii, et tipud kataksid kogu kuju. Hulknurga annoteerimist kasutatakse pildi segmenteerimise ülesande jaoks.

Annoteerimist vajavaid andmeid on palju ning iga objekti kuju täpikestega manuaalne tähistamine võtaks ebamõistlikult palju aega. Selle probleemi vältimiseks kasutati *hasty.ai*¹ keskkonda, mis võimaldab objekte annoteerida mõne klikiga, kasutades selleks tehisintellekti. *Hasti.ai* keskkond toetab annotatsioonide eksportimist mitmetesse üldtuntud formaatidesse. Käesoleva töö puhul kasutati annotatsioonide jaoks COCO JSON² formaati, mille näide on nähtav allpool (joonis 17). Iga objekti annotatsioon sisaldab viidet pildile ning pildi kategooriale, lisaks on igal annotatsioonil oma ID. Segmentatsiooni maski koordinaadid on tihti salvestatud RLE (*run-length encoding*) formaadis, mis on lihtne kodeerimisskeem, mis teostab kadudeta andmete kompressimist [24].

¹ <https://help.hasty.ai/>

² <https://cocodataset.org/#format-data>

```

annotation{
  "id"           : int,
  "image_id"    : int,
  "category_id" : int,
  "segmentation" : RLE or [polygon],
  "area"        : float,
  "bbox"        : [x,y,width,height],
  "iscrowd"     : 0 or 1,
}

categories[
  {
    "id"           : int,
    "name"         : str,
    "supercategory" : str,
  }
]

```

Joonis 17. COCO JSON formaadi näide

Kuna käesoleva töö esmane eesmärk oli tuvastada pildilt metsloom olenemata tema liigist, siis esmase annoteerimise käigus loodi ainult üks kategooria *Animal*. Hilisemas tööfaasis lisati juurde järgmised kategooriad: karu, põder, metskits, rebane, jänes, inimene.

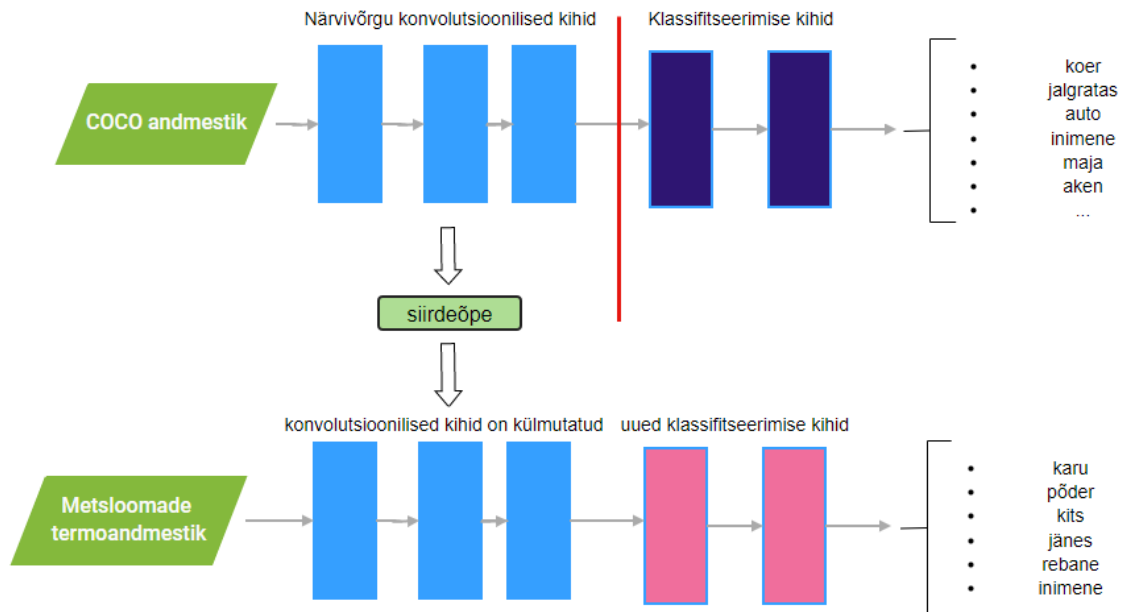
Manuaalne annoteerimine on väga ajakulukas tegevus ning seetõttu on treenimis- ning valideerimisandmestikus minimaalne arv andmeid.

YOLOv5 mudeli treenimiseks oli vajalik andmed konverteerida COCO JSON formaadist YOLOv5 PyTorch formaati. Selleks kasutati RoboFlow¹ tööriista, mis võimaldab importida olemasolevad annotatsioonid ning seejärel need vajalikus formaadis eksportida.

3.3 Erinevate närvivõrkude rakendamine

Käesoleva töö raames loodud annoteeritud loomade termopiltide andmekogu on konvolutsiooniliste närvivõrkude treenimiseks liiga väike. Selle probleemi lahendamiseks otsustati rakendada siirdeõpet (*transfer learning*), mis on tehnika, kus eelnevalt suurte andmekogude peal treenitud mudelite kaale kasutatakse uue mudeli treenimisel. Siirdeõppe meetodi puhul kantakse eeltreenitud mudeli esimesed konvolutsioonilised kihid koos kaaludega üle, viimane ehk kategoriseerimist teostav kiht treenitakse uue andmestikuga (joonis 18). [25]

¹ <https://docs.roboflow.com>



Joonis 18. Käesoleva probleemi jaoks rakendatud siirdeõppe skeem

Konvolutsiooniliste närvivõrkude mudelid pakuvad mitmed masinõppe teegid. Objekti tuvastuseks kasutati Detectron2 ning PyTorch¹ teeke. Detectron2 on *Facebooki AI Research*'i järgmise põlvkonna teek, mis pakub tipptasemel tuvastamis- ja segmenteerimisalgoritme [26]. Detectron2 toetab üle 80 eel-treenitud mudeli, mis toetavad nii objekti tuvastust, semantilist segmenteerimist, isendi segmenteerimist kui ka panoptilist segmenteerimist. Kõik mudelid on eelnevalt treenitud väga suurte andmekogude peal nagu MS-COCO, Cityscapes, LVIS, PASCAL VOC [12]. PyTorch on süvaõppe raamistik, mis pakub tööriistu erinevate närvivõrkude loomiseks ja treenimiseks. See on tasuta ning avatud lähtekoodiga teek.

Kogu tehniline lahendus koos koodiga on leitav Githubist². Lahenduse loomisel on aluseks võetud Detectron2 dokumenteeritud näited³ ning YOLOv5 dokumenteeritud näited⁴ Google Colab keskkonnas.

¹ <https://pytorch.org/>

² <https://github.com/triinruutli/wild-animal-detection>

³ https://colab.research.google.com/drive/16jcaJoc6bCFAQ96jDe2HwtXj7BMD_-m5

⁴ <https://colab.research.google.com/github/roboflow-ai/yolov5-custom-training-tutorial/blob/main/yolov5-custom-training.ipynb>

Enne närvivõrgu üle treenimist valdkonnaspetsiifiliste andmetega on vajalik laadida need andmed õiges formaadis nii, et need oleksid Detectron2 jaoks töödeldavas vormis. Sarnaselt COCO annotatsioonide formaadile laetakse andmed sisse sõnastike (*dictionary*) listina, kus iga kirje sisaldab informatsiooni ühe pildi kohta. Olenevalt tehisnägemise ülesandest võivad kohustuslikud väljad sõnastikus erineda, kuid põhistruktuuri kuuluvad väljad nagu faili nimi, kõrgus, laius, faili ID, annotatsioonid. Annotatsioonid koosnevad samuti sõnastike listist, kus iga sõnastiku kirje on COCO JSON'iga sarnases formaadis, koosnedes järgmistest väljadest: kategooria ID, piirdekasti (*bounding box*) koordinaadid, segmentatsiooni mask. Kui annotatsioonide list on tühi, siis ei ole sellel pildil ühtki vajalikku objekti ning need pildid arvatakse automaatselt treenimisel välja. Semantilise segmenteerimise ülesande jaoks on lisaks annotatsioonidele tarvis ka põhitõe (*ground truth*) halltoonides pildifaili, kus pildipikslitel on küljes kategooria ID. Panoptilise segmentatsiooni jaoks on samuti vajalik põhitõe (*ground truth*) fail, kus RGB pildi pikslitel on kategooria ID. Erinevalt semantilisest segmenteerimisest on siinkohal oluline lisainformatsioon kategooriate kohta, mida hoiustatakse samuti eraldi sõnastikena. [27]

Käesolevas töös on annotatsioonid juba eksporditud COCO JSON formaadis. Kuna detectron2 teek toetab seda formaati vaikimisi, siis on andmestiku registreerimiseks kasutatud teegi poolt pakutavat funktsiooni *register_coco_instances*. Nimetatud funktsiooni parameetriteks on andmestiku nimi, metadata, viide annotatsiooni json failile ning viide pildifailide kaustale. Tulemuseks on registreeritud andmestik, millega hakatakse närvivõrgu mudelid treenima. [27]

YOLO5v mudeli treenimiseks ei ole vaja luua spetsiifilises formaadis andmekogusid.

3.3.1 Närvivõrkude treenimine

Loomade andmestikul rakendati nii objekti tuvastust piirdekastidega kui ka isendi tuvastust segmenteerimismaskidega. Isendi segmenteerimise ülesande jaoks on kasutatud Mask R-CNN närvivõrku ning objekti tuvastuseks Fast R-CNN ja YOLOv5 närvivõrke. Detectron2 pakutavate mudelite hulgas on 50 ning 101 konvolutsioonikihiga närvivõrke, mis jagunevad veel omakorda põhiarhitektuuri järgi. Mida sügavam ehk mida rohkem konvolutsioonilisi kihte on närvivõrgul, seda täpsemad on küll tulemused, kuid aina keerulisem on närvivõrku treenida. Kõik närvivõrgud on eelnevalt treenitud COCO andmestikul [28], milles on üle 80 erineva objekti kategooria ning 330 000 pildifaili.

Käesoleva töö probleemi lahendamiseks on vajalik mudel üle treenida loomade andmestikuga. Mudeli treenimisel on oluline valida andmestikule ja mudelile sobivad hüperparameetrid. Kuna tegemist on varasemalt treenitud mudelitega, siis võetakse aluseks juba olemasolev konfiguratsioon ning muudetakse ainult vajalike parameetrite väärtusi. Mask R-CNN ning Fast R-CNN mudelite treenimiseks on muudetud kategooriate arvu, õpisammu, iteratsioonide arvu ning treeningandmestike parameetrite väärtusi, et treenimine oleks kohandatud just käesoleva probleemi jaoks. Õpisamm peab olema optimaalne, et leida treenimisel minimaalne kadu. Õpisamm kontrollib parameetrite värskendamise ulatust iga iteratsiooni korral [29]. Iga iteratsiooni käigus kasutatakse kaofunktsiooni, et mõõta ennustuse kõrvalekallet põhitõest ning värskendada mudeli parameetreid eesmärgiga minimeerida kadu [29].

YOLOv5 mudeli treenimiseks on sisendparameetrid järgmised: sisendpildi suurus, partii (*batch*) suurus, epohhide arv, andmete asukoht, kaalud. YOLO5v mudeli puhul on sisendpildid ruudukujulised ning pildi suuruseks on vaikumisi 640x640 pikslit [30]. Kuna loomade andmestikus olevad pildid on suuruses 480x360, siis on sisendpildi suuruse parameetri väärtuseks määratud 480, mis tähendab, et pildi laius on sama, kuid kõrgusele on lisatud halltoonides tühi ruum. Andmete parameeter viitab andmete asukoha aadressile, kust mudel võtab treenimiseks sisendpildid. Kaaludeks võetakse YOLOv5 mudeli vaikumisi kaalud, mis on saadud varasemalt suurte andmemahtude peal treenides. Partii suurus (*batch size*) määrab mitu pilti treenitakse ühe iteratsiooni jooksul. Epohhide arv määrab treeningtsükli arvu kogu andmestikuga ehk mitu korda saadetakse andmekogu läbi närvivõrgu. Partii suurus ning epohhide arv on parameetrid, mida on eksperimenteerimise teel muudetud vastavalt mudeli treenimise tulemustele.

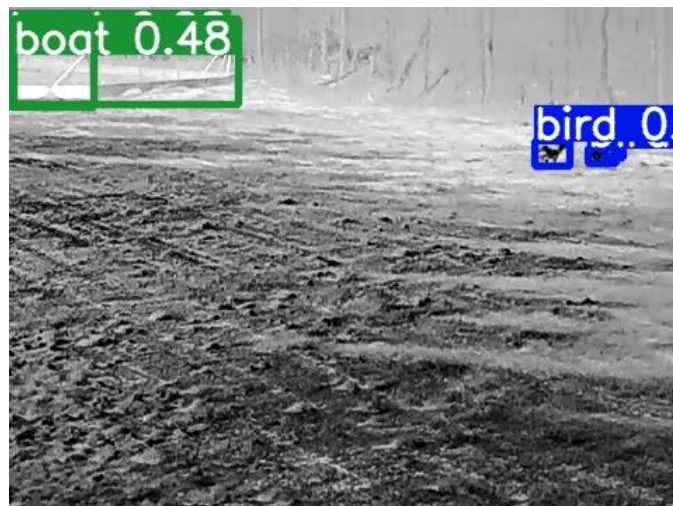
Närvivõrkude treenimise tulemuste hindamiseks kasutati AP mõõdikuid. Nende hulgas on üldine AP näitaja, mis leiab keskmise tuvastustäpsuse üle kategooriate ning objektide suuruste [28]. AP_s näitaja leiab keskmise tuvastustäpsuse väikeste objektide puhul ning AP_m keskmise suurusega objektide puhul. Objekti loetakse väikeseks, kui selle segmentatsioonimaskis olevate pikslite arv on väiksem kui 32^2 (*piirkond* $< 32^2$) [28]. Objekti loetakse keskmiseks, kui selle segmentatsioonimaskis olevate pikslite arv jääb 32^2 ning 96^2 vahele ($32^2 < \text{piirkond} < 96^2$) [28]. Lisaks on spetsiaalne näitaja AP_l suurte objektide jaoks, mille piirkond on suurem kui 96^2 (*piirkond* $> 96^2$) [28]. Käesoleva töö

raames loodud loomade andmestikus ei ole ühtki suurt objekti, seega AP_l moodsikut antud töös hindamiseks ei kasutata.

4 Mudelite treenimine ja valideerimine

Käesolevas peatükis antakse ülevaade süvaõppe mudelite treenimisest ja dokumenteeritakse protsessi käigus saadud tulemused.

Igapäevaste objektide ning RGB piltidega treenitud närvivõrgu mudelid ei anna suure tõenäosusega käesoleva töö probleemi lahendamiseks soovitud tulemusi. Selle hüpoteesi tõestamiseks rakendati esmalt kaht erineva ülesandega närvivõrku YOLOv5 ning Mask R-CNN loomade termopiltide andmestikul. YOLOv5 mudeli puhul tagastati nii valesti kategoriseeritud objekte kui ka valetuvastusi. Mask R-CNN mudel suutis üldjuhul objekti lokaliseerida, kuid kategoriseerimisel arvati tihti, et tegemist on inimese, koera või linnuga. Kvalitatiivsed tulemused on nähtavad allolevatel joonistel 19 ning 20.



Joonis 19. Treenimata YOLOv5 mudeli rakendamine loomade termoandmestikul

Väljundpiltidel on lisaks tuvastatud piirdekastidele näha ka numbrilisi väärtusi. Need väärtused näitavad mudeli poolt pakutavad tõenäosust ennustatud objekti kategooria kohta.



Joonis 20. Treenimata Mask R-CNN mudeli rakendamine loomade termoandmestikul

Vajadust mudeleid esmalt treenida probleemispetsiifiliste andmetega on näha ka alternatiivse andmestiku salvestusfailidelt, kus pildil on juba rakendatud termokaamera tootja poolt pakutud seni parimat objektituvastuse algoritmi. Saadud tulemustelt on näha, et tuvastamine on väga hektiline ja pole usaldusväärne. Lisaks loomadele on mudel tuvastanud ka autosid, teeäärseid poste ning piirdeid. Loomade tuvastamise probleemi raames on need objektid ebavajalikud ning nende lokaliseerimine mudeli poolt lisab vaid tarbetut keerukust. Treenimata mudeli rakendamise tulemused on viidatud lisa 2.

4.1 Loomade termopiltide annoteeritud andmekogu loomine

Käesoleva uurimistöö ühe osana loodi metsloomade termopiltide andmekogu, mis koosneb originaalpiltidest ning nende piltide annotatsioonidest. Kuna avalikult kättesaadavad termopiltide andmestikud on piiratud, siis on mainitud andmekogu loomise eesmärk pakkuda närvivõrkude treenimiseks sobivaid termoandmeid.

Salvestusfailide manuaalsel läbi töötamisel keskenduti videotele, kus liikuv loom on kaamerapildis nähtav ning loomaliik eristatav. Algandmete hulgas on väga palju videofaile, kus aktiivset looma tegutsemas ei ole ning salvestatakse tühja maastikku või ainult autode liikumist. Kogu ületusala peale on salvestamas ainult üks termokaamera, seetõttu on salvestatud ka liikuvaid objekte, mis on väga kaugel ning mille puhul on loomaliiki visuaalsel hindamisel keeruline õigesti kategoriseerida. Selliseid salvestisi andmekogusse ei lisatud. Esimese töötlemise tulemusena tekkis 45 videofaili, millel on näha aktiivset loomade liikumist teeäärsetel aladel või teeületuspiirkonnas. 45 videofaili sisu on eksporditud .JPG formaadis piltideks.

Testandmekogu loomiseks oli saadud andmestik endiselt liiga suur ning koosnes paljudest ebavajalikest piltidest, millel loom on kaamerapildist mõneks hetkeks välja jäänud. Kuna mõnel videol liiguvad loomad väga aeglaselt, siis on eksportides tekkinud palju pilte, mis on praktiliselt samasuguse sisuga. Sellised pildid treenimisele palju kaasa ei aita, seega kõiki loomadega pilte andmekogusse ei lisatud. Seega vähendati ka sama looma piltide osakaalu andmestikus.

Eksportitud pildifailide nimed koosnesid ajatemplist, mis sisaldas viidet ajatsoonile, lisades faili nime lõppu +03. Kuna + sümbol failinimes tekitas treenimisel probleeme, siis eemaldati kõikide eksportitud piltide failinimedest ajatsooni viide. Selleks kasutati arvuti käsuraal *REN* käsku, mis muudab vajalikus kataloogis olevate failide nimesid.

Joonisel 21 on näide treeningandmekogu pildist ning joonisel 22 annoteeritud versioon samast pildist.



Joonis 21. Näide treeningandmekogu pildist



Joonis 22. Näide annoteeritud treeningandmekogu pildist

Andmete annoteerimisel loodi järgmised kategooriad: karu, põder, kits, rebane, jännes, inimene. Inimese kategooria lisati eesmärgiga tulevikus eristada kaamerapildilt inimesi ning selle alusel rakendada teistsugust käitumisviisi kui looma puhul (näiteks ei käivitata inimese puhul alarmi). Andmete puhastamise ning annoteerimise tulemusena tekkisid treening- ning valideerimisandmekogud. Treenimisandmekogus on 220 annoteeritud pildifaili, millest 36 objekti kuulub rebase, 82 kitse, 23 põdra, 81 karu, 32 jänese ning 7 inimese kategooriasse. Valideerimisandmestik koosneb 100 annoteeritud pildifailist, millest 27 objekti kuulub rebase, 37 kitse, 20 põdra, 8 karu, 15 jänese ning 6 inimese kategooriasse.

4.2 Loomade tuvastamine segmenteerimise teel

Käesoleva töö algusetapis kaalus autor objekti tuvastuseks kaht lähenemist: piirdekastid ning pildi segmenteerimine. Esmalt rakendati närvivõrke ainult ühe kategooriaga (*Animal*) annoteeritud andmekogul, tehes kindlaks, kas pildil on loom või mitte. Keerukuse lisamiseks ning lisainformatsiooni saamiseks annoteeriti pildid 6 erineva kategooriaga, et teha kindlaks, milline loomaliik on pildil. Käesolevas alapeatükis kirjeldatakse erinevate meetodite rakendamisel saadud tulemusi.

Isendite segmenteerimise jaoks valiti R50-FPN Mask R-CNN mudeli arhitektuur, mis on näidanud väga häid tulemusi RGB piltide segmenteerimisel ja otsustati see kohandada metsloomade tuvastamiseks termokaamera piltidelt. Kuivõrd termopiltide andmestik on väike, siis otsustati kasutada siirdeõpet ning laeti COCO andmestikul eeltreenitud R50-FPN Mask R-CNN mudel, mis on kättesaadav Detectron2 teegi kaudu. Seega võeti treenimisel aluseks mudeli RGB piltide jaoks treenitud mudeli kaalud, mis on saadud COCO andmestikul varasemast treenimisest.

4.2.1 Ühe kategooriaga annoteeritud andmestikul treenimine

Ühe kategooriaga annoteeritud testandmestiku puhul configureeriti mudel käesoleva probleemi jaoks. Valitud parameetrid on näha tabelis 1. Mainitud andmestiku jaoks osutus optimaalsemaks muuta treeningpiltide arvu iteratsiooni kohta, õpisammu, iteratsioonide arvu ning kategooriate arvu treeningandmestikus. Hüperparameetrite jaoks on loodud konfiguratsiooni fail, kus sätestatakse treenimiseks vajalikud parameetrid ning nende väärtused.

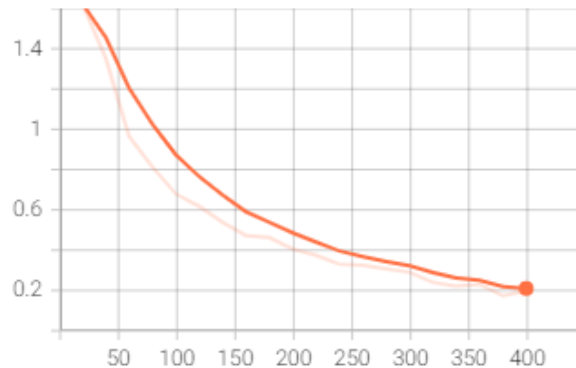
Parameeter	Väärtus
Segmentatsioonimaski formaat	bitmask
Õpisamm	0.0003
Iteratsioonide arv	400
Klasside arv	1

Tabel 1. Ühe kategooriaga annoteeritud testandmestiku treenimiseks kohandatud hüperparameetrid

Treenimise tulemuste visuaalseks esituseks kasutatakse Tensorboard rakendust, mis on TensorFlow¹ masinõpperaamistiku poolt välja arendatud tööriist. Tensorboard

¹ <https://www.tensorflow.org/>

graafikutelt saab välja lugeda, et x -teljel on 400 iteratsiooniga Mask R-CNN mudeli treenimisel saavutatud y -teljel minimaalne kadu 0.196 ühiku juures (joonis 23).



Joonis 23. Mask R-CNN mudeli treenimise kao graafik

Mudeli treenimise tulemusena saadud kaalud (*weights*) salvestatakse .PTH formaadis faili ning neid kasutatakse valideerimisel. Tulemuste valideerimise käigus rakendatakse äsja treenitud närvivõrku valideerimisandmestikul. Valideerimisandmestik koosneb andmetest, mis on samatüübilised, kuid mitte identsed ja mida ei ole kasutatud treenimisel. Valideerimistulemuste järgi saab hinnata mudeli üldistusvõimet uute andmete peal. Joonisel 24 on juhuslikult selekteeritud pilt valideerimisandmestikust, millel on rakendatud treenitud närvivõrku. Lisas 3 on videofail, millel on rakendatud treenitud närvivõrku.



Joonis 24. Treenitud mudeli rakendamisel saadud tulemus

Kui hinnata tulemusi AP mõõdikute alusel, siis üldine keskmine täpsus AP on 71%. Kuna väiksemate objektide keskmine tuvastustäpsus on 66%, siis see vähendab ka üldise AP

mõõdiku täpsusprotsenti. Keskmise suurusega objektide tuvastustäpsus AP_m on 85%. Keskmise saagis on 74%, mis näitab, kui suur osa kõikidest andmestikus annoteeritud objektidest on õigesti kategoriseeritud ning lokaliseeritud.

4.2.2 Mitme kategooriaga annoteeritud andmestikul treenimine

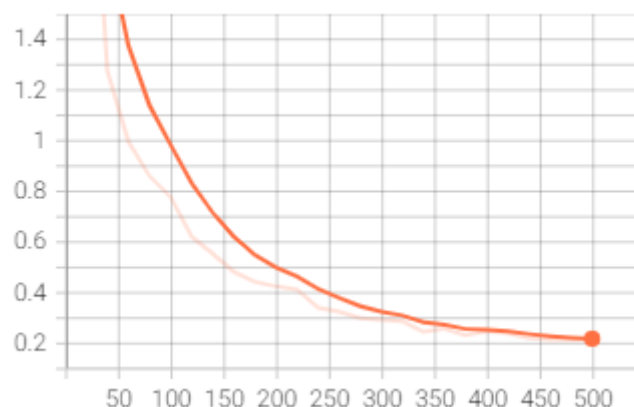
Kuna ühe kategooriaga annoteeritud andmekogul valideerimine andis oodatust paremaid tulemusi, siis otsustati keerukuse lisamiseks ning lisainformatsiooni kogumiseks kasutada rohkem kui üht kategooriat objektide tuvastamiseks.

Mitme kategooriaga andmestiku treenimiseks on taaskord vajalik muuta hüperparameetreid. Õpisammu ning iteratsioonide arvu muudeti vastavalt eksperimenteerimise teel saadud tulemustele. Tabelis 2 on kirjeldatud muudetud parameetrid ning nende väärtused.

Parameeter	Väärtus
Segmentatsioonimaski formaat	bitmask
Treeningpiltide partii suurus	2
Õpisamm	0.0015
Iteratsioonide arv	500
Klasside arv	6

Tabel 2. Mitme kategooriaga annoteeritud testandmestiku treenimiseks kohandatud hüperparameetrid

Joonisel 25 on näha kaofunktsiooni kõverat, mille y -telg iseloomustab kadu ning x -telg iteratsioonide arvu.



Joonis 25. Treenitud Mask R-CNN mudeli kaofunktsiooni kõver

Mudeli treenimise tulemusena saadud kaalud (*weights*) salvestatakse .PTH formaadis faili ning neid kasutatakse valideerimisel. Joonisel 26 on juhuslikult selekteeritud pilt valideerimisandmestikust, millel on rakendatud treenitud närvivõrku ning tagastatud on

õigesti lokaliseeritud ja kategoriseeritud objektid. Joonisel 27 on näha õigesti lokaliseeritud, kuid valesti kategoriseeritud objekti. Lisas 4 on videofail, millel on rakendatud treenitud närvivõrku.



Joonis 26. Mask R-CNN mudeli rakendamisel saadud õigesti lokaliseeritud ning kategoriseeritud objektid



Joonis 27. Mask R-CNN mudeli rakendamisel saadud õigesti lokaliseeritud, kuid valesti kategoriseeritud objekt

Valideerimisel kasutati täpsuse hindamiseks AP mõõdikuid. Keskmise täpsus AP on 47%, väikeste objektide puhul 33% ning keskmiste objektide puhul 57%. Keskmise saagis on 50%.

4.3 Objektide tuvastamine piirdekasti meetodil

Võrdluseks isendi segmenteerimisele otsustati rakendada ka piirdekastide meetodil objektide tuvastamist. Selleks kasutati kaht erineva arhitektuuriga närvivõrku: Fast R-CNN R50FPN ning YOLOv5. Kuna andmed on annoteeritud COCO JSON formaadis, siis see formaat sisaldab nii koordinaate piirdekastide kui ka segmentatsioonimaski jaoks, seega andmeid kummagi meetodi jaoks eraldi annoteerida ei ole vaja. Küll aga konverteeriti YOLOv5 mudeli jaoks annotatsioonid ümber COCO JSON formaadist YOLOv5 PyTorch formaati.

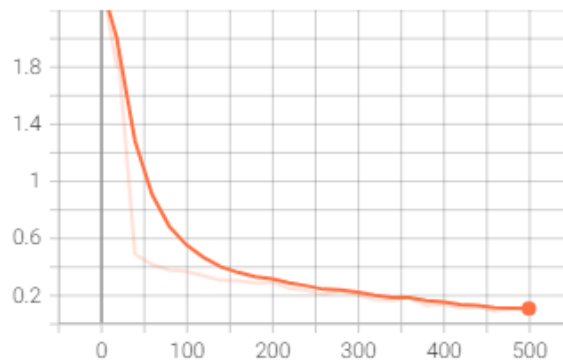
4.3.1 Fast R-CNN mudeli treenimine

Sarnaselt Mask R-CNN mudeli treenimisele võeti ka Fast R-CNN mudeli treenimiseks aluseks varasemalt COCO andmestikul treenitud kaalud. Tabelis 3 on kirjeldatud muudetud hüperparameetrid ning nende väärtused.

Parameeter	Väärtus
Segmentatsioonimaski formaat	polygon
Treeningpiltide partii suurus	2
Õpisamm	0.002
Iteratsioonide arv	500
Klasside arv	6

Tabel 3. Fast R-CNN mudeli treenimiseks kohandatud hüperparameetrid

Joonisel 28 on kaofunktsiooni kõver, mis iseloomustab treenimise käigus leitud minimaalset kadu.



Joonis 28. Treenitud Fast R-CNN mudeli kaofunktsiooni kõver

Joonisel 29 on juhuslikult selekteeritud pilt valideerimisandmestikust, millel on rakendatud treenitud närvivõrku ning tagastatud on õigesti lokaliseeritud ja kategoriseeritud objekt. Joonisel 30 on näha õigesti lokaliseeritud, kuid valesti kategoriseeritud objekt.



Joonis 29. Fast R-CNN treenitud mudeli õigesti lokaliseeritud ja kategoriseeritud tulemus



Joonis 30. Fast R-CNN treenitud mudeli õigesti lokaliseeritud, kuid valesti kategoriseeritud tulemus

Mudeli tulemuste hindamisel saadi AP mõõdikuks 39%, mis on väga madal tulemus. Väiksemate objektide tuvastusprotsent oli 34% ning keskmiste objektide tuvastustäpsus

37%. Tulemuste hindamisel kategooriate järgi AP mõõdikutega saadud tulemused on tabelis 1.

Kategooria	AP
Fox	32.934
Bear	38.995
Deer	39.541
Rabbit	0.000
Moose	78.839
Human	0.000

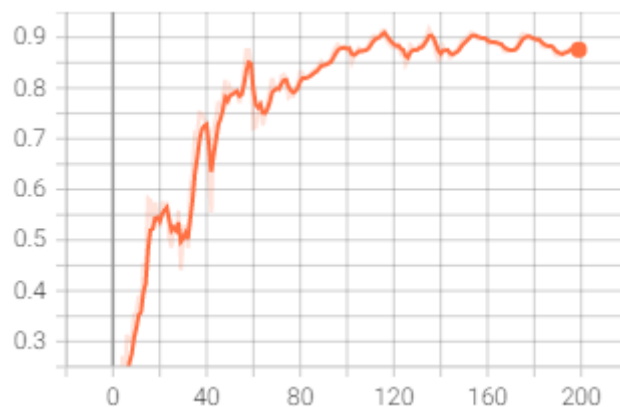
Tabel 4. Fast R-CNN mudeli treenimise tulemuste hindamine kategooriate järgi

Keskmine saagis AR (*average recall*) on 42%, mis näitab, kui palju on tagastatud õigesti lokaliseeritud ning kategoriseeritud objekte koguhulgast.

4.3.2 YOLOv5 mudeli treenimine

Detectron2 teek ei toeta YOLO mudeleid, seetõttu kasutati YOLO mudeli treenimiseks PyTorch'i teeki, mis võimaldab valida erinevate YOLO mudelite vahel [31]. Andmete lugemiseks kasutati RoboFlow tööriista.

Mudeli treenimisel katsetati erinevaid epohhide arve ning treeningpiltide partii suuruseid. Lõplik tulemus on treenitud 200 epohhiga ning partii suuruseks valiti 16 pilti. Keskmine täpsus treenimisel on 90%, graafikut on näha joonisel 31. Mudeli täpsus (*precision*) ning saagise (*recall*) mõõdikud tagastasid vastavalt 93% ning 83%.



Joonis 31. YOLO mudeli treenimisel saadud mAP

YOLOv5 mudeli valideerimisel saadud tulemused on nähtaval joonisel 32, kus on näide õigesti kategoriseeritud ning lokaliseeritud ennustusest ning joonisel 33, kus on näide

valesti kategoriseeritud, kuid õigesti lokaliseeritud ennustusest. Lisas 5 on välja toodud ka YOLOv5 treenitud mudeli tulemused videofailil.



Joonis 32. YOLO mudeli õnnestunud ennustus



Joonis 33. YOLO mudeli ebaõnnestunud ennustus

5 Analüüs ja järeldused

Käesolevas peatükis analüüsitakse saadud tulemusi ning vastatakse uurimisküsimustele. Uurimustöö on läbi viidud Eestis kasutusel oleva loomatuvastussüsteemi parendamise perspektiivist, seega pakutakse peatükis soovitusi ning juhiseid võimalike muudatuste tegemiseks. Lisaks tuuakse välja uurimustöö edasiarenduse võimalused.

5.1 Kuidas ning milliseid süvaõppe meetodeid kasutades saab tuvastada metsloomi termopiltidelt?

Nagu eelmises peatükis uuriti, on objekti tuvastuseks mitmeid meetodeid. Olenevalt tuvastamise ülesandest on võimalik objekti kategoriseerida ning lokaliseerida piirdekastikese abil kui ka pikslite tasemel ehk segmenteerimise abil. Kummagi ülesande jaoks rakendatud närvivõrgud Mask R-CNN, Fast R-CNN ning YOLOv5 on väga populaarsed ning laialdaselt kasutatud närvivõrgud. Kuigi mainitud närvivõrgud on varasemalt treenitud RGB piltidega, on kõik kolm närvivõrku käesoleva töö raames näidanud häid tulemusi ka termopiltidega treenimisel. Allolevas tabelis (Tabel 5) on välja toodud treenitud närvivõrkude tulemused.

Närvivõrk	Keskmine täpsus (AP)	Valideerimise kiirus
Mask R-CNN - ühe kategooriaga annoteeritud andmekogul	71%	28 s kogu andmestik, 0.319 s pildi kohta
Mask R-CNN - mitme kategooriaga annoteeritud andmekogul	47%	27 s kogu andmestik, 0.320 s pildi kohta
Fast R-CNN - mitme kategooriaga annoteeritud andmekogul	31%	28 s kogu andmestik, 0.353 s pildi kohta
YOLOv5 - mitme kategooriaga annoteeritud andmekogul	90%	18 s kogu andmestik, 0.022 s pildi kohta

Tabel 5. Närvivõrkude treenimise tulemused

Võrreldes tabelis 5 välja toodud AP väärtusi on YOLOv5 näidanud ülekaalukalt parimaid tulemusi nii täpsuses kui ka valideerimise kiiruses. Valideerimise kiirus on mudelite hindamisel väga oluline mõõdik, kuna mudelit plaanitakse hakata rakendama videofailidel, kus pildid vahetuvad väga kiiresti ning tuvastamine peab samuti toimuma kiiresti. Samuti on häid tulemusi näidanud ühe kategooriaga annoteeritud andmestikul treenitud Mask R-CNN mudel, mis on ka oodatav, sest võrreldes teiste mudelitega on siinkohal tuvastamiseks ainult üks klass. Fast R-CNN on nii täpsuses kui ka kiiruses näidanud kehvaid tulemusi ning see mudel käesoleva probleemi lahendamiseks ei sobi. Võrreldes YOLOv5 mudeliga on 1 kategooriaga Mask R-CNN mudeli valideerimiskiirus palju aeglasem, olles pigem võrdväärne mitme kategooriaga Mask R-CNN ning Fast R-CNN mudelite tulemustega. Seega puhtalt tulemustest lähtuvalt on kõige parem lahendus kasutada käesoleva probleemi jaoks YOLOv5 mudelit.

Isendi segmenteerimiseks mitme kategooriaga oleks vajalik suurendada treening- ja valideerimisandmekogudes annoteeritud piltide hulka, et treenimisel oleks rohkem informatsiooni. Samuti on variant olemasolevaid andmeid augmenteerida ehk lisada muudetud kujul koopiaid. Näiteks on augmenteerimise võimalusteks pildi lõikamine, ümber pööramine, peegelpilti muutmine, suuruse muutmine.

5.2 Kuidas mõjutavad erinevad ilmastikutingimused termopiltidel objekti tuvastust?

Termokaamera salvestiste läbi töötamisel andis autor andmetele esmase hinnangu. Selgus, et päevasel ajal on liikumas tunduvalt vähem loomi ning nende märkamine kaamerapildilt on kordades keerulisem kui öisel ajal. Kvalitatiivse valideerimise käigus selgus, et päikesepaistelisel päeval ei ole looma termojälg nähtav nii kontrastselt kui seda on öösel, kui ümbritsev keskkond on jahedam ning päiksest tekitatud varje maastikul on vähem. Seetõttu on andmestikus rohkem pilte hämaras või pimedal ajal salvestatud loomadest. Joonisel 34 on näha termokaamera salvestatud pildifaili kitsedest päikesepaistelisel päeval.



Joonis 34. Termokaamera pilt kitsedega päikesepaistelistes tingimustes

Andmestikus on ka faile, mis on salvestatud nii vihmastes kui ka udustes ilmastikutingimustes. Mõlema situatsiooni puhul on looma termojälg termokaamera pilti edukalt üle kandunud ning loom on nähtav ja eristatav ümbritsevast keskkonnast (joonis 35). Lisas 6 on videofail, mis on salvestatud vihmastes ilmastikutingimustes ning millel on rakendatud treenitud närvivõrku.



Joonis 35. Termokaamera pilt vihmastes tingimustes

Samuti ei ole probleemi loomade tuvastamisel lumistes tingimustes. Looma termojälg kantakse edukalt üle termokaamera pildile, nagu võib näha jooniselt 36. Lisas 6 on videofail, mis on salvestatud lumistes ilmastikutingimustes ning millel on rakendatud treenitud närvivõrku.



Joonis 36. Termokaamera pilt lumistes tingimustes

Kuna närvivõrgud on treenitud annoteeritud andmekogudel, mis sisaldavad pilte päikeselistes, vihmastes, udustes ning lumistes oludes, siis suudab ka närvivõrk tuvastada objekti otseselt sõltumata ilmastikutingimustest. Päikeselistes oludes olenevalt maastikust võib tuvastamise täpsus olla kehv, kuna närvivõrk võib loomade asemel päikesest tekkivate varjude tumedaid kogusid tuvastada (joonis 34). Lisaks võib üks võimalik probleem olla suvistel perioodidel soojuste peegeldumine asfaltkattega maanteelt, mis omakorda raskendab objekti tuvastamist termokaameralt maantee läheduses. Käesoleva töö raames välja töötatud loomade andmekogus ei ole kahjuks pilte, mis oleks tehtud tugeva lumesaju ajal, seega siinkohal ei ole teada, kas ja kui täpselt suudab närvivõrk objekti sellistes tingimustes tuvastada.

5.3 Kas tuvastamise täpsus langeb kauguse suurenedes?

Toetudes eelmises peatükis leitud tulemustele, on kauguse suurenedes väiksemate objektide tuvastamise täpsus aina väiksem. Tuvastuse täpsust on hinnatud COCO mõõdikutega AP_s ning AP_m , mis tähistavad täpsust väikeste ning keskmiste objektide kohta. Loomade andmekogus ei ole objekte, mis oma mõõtmetelt kuuluksid COCO suurte objektide kategooriasse [28]. Mainitud mõõdikud on iga mudeli puhul erinevad, väärtused mudelite kaupa on nähtavad alltoodud tabelis 6. Yolo mudeli puhul ei olnud võimalik kasutada AP mõõdikuid väikeste ning keskmiste objektide mõõtmiseks, seega selle mudeli puhul on hinnatud tulemusi kvalitatiivselt.

Närvivõrk	AP_s	AP_m
Ühe kategooriaga annoteeritud andmestikul treenitud Mask R-CNN	66%	82%
Mitme kategooriaga annoteeritud andmestikul treenitud Mask R-CNN	34%	57%
Fast R-CNN	22%	32%

Tabel 6. Väikeste ja keskmiste objektide tuvastustäpsus

Täpsuse vähenemise põhjuseks kauguse suurenemisel võib olla väike treeningandmestik, kus kaugemal asuvate objektidega pilte on vähe, kuid kindlasti on oluline märkida, et kaugemate objektide puhul on väga raske ka visuaalsel hinnangul looma liiki õigesti kategoriseerida. Tihti on kauguses näha vaid musta kogu ning see objekt võib kuuluda ükskõik millisesse loomaliiki. Termokaamera on asetatud nii, et kaamerapilt kataks kogu teeületuspiirkonna, mille pindala on 3,6 ha, seega kaugematel aladel liikuvad loomad ei jää tihti peale tuvastamiseks piisavalt nähtavalt kaamerasse.

Suurte ulukite tuvastamisprotsent on tunduvalt kõrgem kui väikeulukite puhul. Valetuvastuste hulk on suur jäneste ning rebaste kategoriseerimisel, mil tihti määratakse jännes karu kategooriasse ning rebane kitse kategooriasse arvatavasti seetõttu, et andmekogus on andmeid väga vähe isegi siirdeõppe kohta.

Eestis kasutatava loomatuvastussüsteemi puhul on tegemist suurulukile optimeeritud pilvradar perimeetervälve lahendusega, mis ei ole suunatud väiksemate ulukite avastamiseks [5]. Suuremate loomade tuvastamise täpsus on aga mõnevõrra olulisem kui väiksemate loomade puhul, sest loomatuvastussüsteemi eesmärk on eelkõige vältida kokkupõrkeid suurulukitega. Kuna välja töötatud mudel suudab eristada erinevaid loomaliike, siis lisab käesolev töö lisandväärtust väiksemate loomade tuvastamisel. Tuvastamistäpsuse parandamiseks, mida planeeritakse teha edasiste arendustegevuste käigus, on vajalik suurendada jäneste ning rebaste osakaalu treeningandmestikus.

5.4 Ettepanekud loomatuvastussüsteemi parendamiseks

Käesoleva töö tulemustest lähtuvalt sobiks loomade tuvastamiseks teeületusala piirkonnas kõige paremini YOLOv5 mudeli rakendamine termokaameral.

Uurimustöö käigus on loodud loomaliikide põhjal kategooriad, mis aitavad lisaks objekti lokaliseerimisele ka klassifitseerida loomaliiki. Samuti on lisatud inimese kategooria, et

eristada inimese liikumist metslooma omast. Kuna hetkel seisneb termokaamerate kasutamise eesmärk loomatuvasustusteemide osas vaid loomaliikide loendamises ning nende liikide käitumismustrite ennustamises [5], siis aitab liikide kaupa loomade kategoriseerimine kasvõi osaliselt automatiseerida teed ületavate loomade statistika pidamist, vähendades manuaalset tööd videote läbi vaatamisel. Käitumismustrite uurimist ning tuvastust on samuti võimalik lisaarendusena automatiseerida, toetudes tuvastatud kategooriatele. Samuti on võimalik reguleerida vastavalt loomaliigile või looma suurusele liiklusmärkidel kuvatavat informatsiooni. Mida suurem loom ning kiirem liikumine, seda suurem peaks olema ka kiirusepiirang. Liikumise kiirust on võimalik jälgida radarite abil, mis on juba installeeritud olemasolevasse loomatuvasustusteemi [5]. Inimese tuvastamisel on võimalik rakendada teistsugust tegutsemisviisi või vajadusel eirata liikluspääsude edastamist liiklejatele.

Looma tuvastamise meetodi kasutusele võtmisel termokaamerate abil on võimalik jätkata ka ühe pöördtermokaameraga teeületusala kohta, kuid tulenevalt objekti suuruse tuvastamistäpsuse analüüsis leitud tulemustest on kaugemal asuvate objektide puhul täpsus madal ning mudeli kasutusele võtmine ei anna maksimaalselt häid tulemusi. Seetõttu tehakse ettepanek vahetada üks pöördkaamera mitme lainurkkaamera vastu, mille vaateväljad kataksid ära kogu teeületusperimeetri. Lisaks annab lainurkkaamerate kasutamine lisandväärtust olukorras, kus teed on ületamas mitu looma erinevatest kohtadest. Pöördtermokaamera peab sellises olukorras tegema valiku, millist looma jälgida.

Tulemustest lähtuvalt pakutakse siinkohal välja lisaks termokaamerale ka tavalise RGB kaamera kasutamist päevasel ajal. Kuna päevasel ajal on päikesepaistelistes tingimustes objekti tuvastamine tihti keeruline ja valetuvastuste tagastamise tõenäosus on suurem (näide joonisel 34), siis võiks siinkohal töötada RGB kaamerapildilt looma tuvastamine. Termokaamera ning RGB kaamera ühise kasutamise idee seisneb üksteise täiendamises erinevate ilmastikutingimuste ning nähtavuse korral. RGB kaamerapildil väljatöötatud mudeli rakendamise tulemused pole teada, selle välja selgitamine kuulub edasiste arendustegevuste hulka.

Eesti loomaõnnetuste registri loomise uurimuse kohaselt tuvastati riigimaanteedel statistiliselt olulisi loomaõnnetuste koondumiskohti kokku 322,2 km ulatuses, mis on võrdlemisi suur maa-ala, mida on vajalik muuta nii liiklejatele kui ka loomadele

ohutumaks [2]. Uurimistöö käigus välja töötatud mudeli rakendamine termopiltidel oleks võrdlemisi odav lahendus, mida on võimalik kasutusele võtta ka väiksematel maanteedel ilma radariteta. Lisaks termokaameratele tekiks lisakulu arvatavasti muutteabemärkide soetamiseks, mis edastaksid looma tuvastuse korral informatsiooni liiklejatele.

5.5 Edasised arendused

Uurimistöö tulemusena valminud närvivõrgu mudel näitab küll häid tulemusi loomade tuvastamisel, kuid suurema tuvastustäpsuse saavutamiseks on siiski vajalik jätkata arendusega. Nagu mudelite valideerimise tulemustest võib välja lugeda, on väiksemate loomade tuvastustäpsus tunduvalt väiksem kui suuremate loomade puhul. Seetõttu on vajalik suurendada nii jäneste kui ka rebaste osakaalu treeningandmestikus, mis tähendab originaalpiltide sorteerimist ning annoteerimist. Samuti on vajalik isendi segmenteerimise parendamiseks suurendada loomade treeningandmestikku, et mudeli treenimisel oleks võimalikult mitmekesiseid andmeid.

Käesoleva töö raames töötati välja Eesti tingimustes metsloomade termopiltidel tuvastamise ülesande jaoks närvivõrgu mudel, kuid järgmise sammuna on vajalik mudelit rakendada otsepildis termokaameral ning analüüsida sealseid tulemusi. Esialgu ei pea võtma termokaamerat kasutusele kui lisakontrolli lisaks radaritele, vaid valideerimiseks piisab, kui teatud ajaperioodi kohta koguda tulemusi närvivõrgu poolt tagastatud andmete kohta. Mudeli rakendamiseks termokaamera otsepildis on vajalik tegevused kooskõlastada Transpordiameti ning loomatuvastussüsteemi tarnija AS Kiirwarreniga.

Kui välja töötatud mudelit on plaanis kasutada väljaspool Kose-Võõbu maanteel asuvat loomatuvastussüsteemi, siis esmalt peab hindama mudeli sobivust teistsuguses keskkonnas tehtud termopiltidega. Mudeli töökindluse huvides võiks rakendada välja töötatud mudelit ka teistsuguses keskkonnas tehtud piltidel ja analüüsida, millised on saadud tulemused. Vajadusel saab täiendada treening- ja valideerimisandmestikke erinevate termopiltidega ning mudeli taaskord üle treenida täiendatud andmetega.

Töö planeerimisetapis sooviti tööle saada ka panoptiline segmenteerimine termopildil, kuid isendi segmenteerimise tulemusi analüüsides oli selge, et andmestikus ei ole piisavalt andmeid rakendamaks panoptilist segmenteerimist. Tulemuste võrdlemise huvides võiks jätkata peale andmestiku suurendamist ka panoptilise segmenteerimisega.

6 Kokkuvõte

Igal aastal registreeritakse Eesti teedel keskmiselt 3000 sõiduki ja looma kokkupõrget. Üheks loomade hukkamise põhjuseks on suur liiklustihedus. Töö eesmärgiks oli töötada välja konvolutsioonilise närvivõrgu mudel, mis suudaks tuvastada metslooma objekti termopildilt ning luua metsloomade termopiltide andmekogu. Uurimistöö on kirjutatud Eestis kasutatava loomatuvastussüsteemi parendamise perspektiivist. Käesoleva töö väljund annab süsteemile lisandväärtust loomade tuvastustäpsuse suurendamises, rakendades termokaameratel objekti tuvastuseks välja töötatud mudelit.

Konvolutsioonilised närvivõrgud on näidanud väga häid tulemusi piltide analüüsimises ning objektide tuvastamises. Objektide lokaliseerimiseks ning kategoriseerimiseks rakendati nii objekti tuvastust piirdekastidega kui ka isendi tuvastust segmenteerimismaskidega ehk isendi segmenteerimist. Loomade tuvastamiseks termopildilt on kasutatud siirdeõppe meetodit, mille korral on närvivõrgu mudelid üle treenitud probleemspetsiifiliste andmetega. Töö raames on loodud metsloomade termopiltide annoteeritud andmekogu, mille abil on treenitud erinevaid närvivõrgu mudeleid tuvastama teeäärses keskkonnas liikuvaid metsloomi.

Tuvastuseks uuriti 3 erineva arhitektuuriga närvivõrku: Mask R-CNN, Fast R-CNN ning YOLOv5. Mask R-CNN mudelit treeniti esmalt ühe kategooriaga annoteeritud andmekogul, keerukuse lisamiseks otsustati objektid kategoriseerida järgmiselt: karu, põder, rebane, jänes, kits, inimene. Kolmest mudelist näitas parimaid tulemusi nii tuvastamise täpsuses kui ka valideerimise kiiruses YOLOv5, mida soovitatakse ka loomatuvastussüsteemis kasutusele võtta. YOLOv5 näitas rahuldavaid tulemusi ka väikeste objektide tuvastamises, kuid väikeste objektide paremaks tuvastustäpsuseks on vajalik andmekogu täiendada.

Vastavalt töös leitud tulemustele teeb autor ettepaneku loomatuvastussüsteemis välja vahetada pöördmehhanismiga termokaamera mitme lainurkkaamera vastu, et kaugemal asuvaid loomi paremini tuvastada. Lisaks pakutakse välja RGB ning termokaamera kasutamise kombinatsiooni, kuna töö käigus leitud tulemustele tuginedes võib päikesepaistelistel ilmadega olla üksnes termokaameralt objekti tuvastamine raskendatud. Välja töötatud lahendust on võimalik rakendada ka väiksematel maanteedel lihtsama loomatuvastussüsteemina.

Kasutatud kirjandus

- [1] Eesti Transpordiamet, "Elusloodus", 2021. [Online] <https://www.transpordiamet.ee/maanteed-veeteed-ohuruum/keskkonnamoju/elusloodus> (05.04.2022)
- [2] Hendrikson & Ko, "Eesti riigimaanteed võrgu loomaõnnetuste registri täiendamine, liiklusohtrlike lõikude selgitamine ning kaardirakenduse loomine", lõpparuanne, 2019. [Online]. <https://hendrikson.ee/maps/Loomaohtrlikkus/dokumendid/L%C3%B5pparuanne.pdf> (05.04.2022)
- [3] Stratum OÜ, Rewild OÜ, IB Floor/Traffest OÜ, "Suurulukite tuvastussüsteemi teostatavusuuring", 2021.[Online] <https://transpordiamet.ee/media/4469/download> (20.04.2022)
- [4] Eesti Transpordiamet, "Mis ajal satuvad metsloomad enim teedele?", 2022. [Online] <https://www.mnt.ee/et/tee/ulukid/mis-ajal-satuvad-metsloomad-enim-teedele> (19.03.2022)
- [5] Videointervjuu Transpordiameti ITS arendusjuhi Kristjan Duubase ning süsteemi tarnija firma AS Kiirwarren esindaja Andres Nurmega, läbi viidud 23.03.2022
- [6] R. J. Doviak, D. S. Zrnic, and D. S. Sirmans, "Doppler weather radar," *Proceedings of the IEEE*, vol. 67, no. 11, pp. 1522–1553, Nov. 1979, doi: 10.1109/PROC.1979.11511.
- [7] C. Druta, A. S. Alden, "Implementation and Evaluation of a Buried Cable Animal Detection System and Deer Warning Sign", *Virginia Tech Transportation Institute*, 2019. [Online]. https://www.virginiadot.org/vtrc/main/online_reports/pdf/19-r28.pdf (20.04.2022)
- [8] Y. Munian, A. Martinez-Molina, D. Miserlis, H. Hernandez, and M. Alamaniotis, "Intelligent System Utilizing HOG and CNN for Thermal Image-Based Detection of Wild Animals in Nocturnal Periods for Vehicle Safety," *Applied Artificial Intelligence*, pp. 1–29, Feb. 2022, doi: 10.1080/08839514.2022.2031825.
- [9] M. Krišto, M. Ivasic-Kos, and M. Pobar, "Thermal Object Detection in Difficult Weather Conditions Using YOLO," *IEEE Access*, vol. 8, pp. 125459–125476, 2020, doi: 10.1109/ACCESS.2020.3007481.

- [10] C. Li, W. Xia, Y. Yan, B. Luo, and J. Tang, “Segmenting Objects in Day and Night: Edge-Conditioned CNN for Thermal Image Semantic Segmentation,” *arXiv:1907.10303 [cs]*, Jul. 2019. [Online]. <http://arxiv.org/abs/1907.10303> (08.05.2022)
- [11] Y. Li and Y. Zhang, “Application Research of Computer Vision Technology in Automation,” in *2020 International Conference on Computer Information and Big Data Applications (CIBDA)*, Apr. 2020, pp. 374–377. doi: 10.1109/CIBDA50819.2020.00090.
- [12] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, “A Survey of Modern Deep Learning based Object Detection Models,” *arXiv:2104.11892 [cs, eess]*, May 2021. [Online]. <http://arxiv.org/abs/2104.11892> (20.04.2022)
- [13] T.-Y. Lin *et al.*, “Microsoft COCO: Common Objects in Context,” *arXiv:1405.0312 [cs]*, Feb. 2015. [Online]. <http://arxiv.org/abs/1405.0312> (20.04.2022)
- [14] S. Minaee, Y. Y. Boykov, F. Porikli, A. J. Plaza, N. Kehtarnavaz, and D. Terzopoulos, “Image Segmentation Using Deep Learning: A Survey,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2021, doi: 10.1109/TPAMI.2021.3059968.
- [15] A. M. Hafiz and G. M. Bhat, “A survey on instance segmentation: state of the art,” *Int J Multimed Info Retr*, vol. 9, no. 3, pp. 171–189, Sep. 2020, doi: 10.1007/s13735-020-00195-x.
- [16] K. He, G. Gkioxari, P. Dollár, and R. Girshick, “Mask R-CNN,” *arXiv:1703.06870 [cs]*, Jan. 2018. [Online]. <http://arxiv.org/abs/1703.06870> (20.04.2022)
- [17] A. Kirillov, K. He, R. Girshick, C. Rother, and P. Dollar, “Panoptic Segmentation,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 9396–9405. doi: 10.1109/CVPR.2019.00963.
- [18] R. Hou *et al.*, “Real-Time Panoptic Segmentation from Dense Detections,” *arXiv:1912.01202 [cs]*, Apr. 2020. [Online]. <http://arxiv.org/abs/1912.01202> (20.04.2022)
- [19] R. Padilla, S. L. Netto, and E. A. B. da Silva, “A Survey on Performance Metrics for Object-Detection Algorithms,” in *2020 International Conference on Systems,*

- Signals and Image Processing (IWSSIP)*, Jul. 2020, pp. 237–242. doi: 10.1109/IWSSIP48289.2020.9145130.
- [20] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, “The Pascal Visual Object Classes (VOC) Challenge,” *Int J Comput Vis*, vol. 88, no. 2, pp. 303–338, Jun. 2010, doi: 10.1007/s11263-009-0275-4.
- [21] COCO - Common Objects in Context, "Data format" [Online] <https://cocodataset.org/#format-data> (23.04.2022)
- [22] “FREE - FLIR Thermal Dataset for Algorithm Training | Teledyne FLIR.” <https://www.flir.com/oem/adas/adas-dataset-form/> (07.05.2022)
- [23] Make Sense. <https://www.makesense.ai/> (23.04.2022)
- [24] S. Fiergolla and P. Wolf, “Improving Run Length Encoding by Preprocessing,” *arXiv:2101.05329 [cs]*, Mar. 2021. [Online]. <http://arxiv.org/abs/2101.05329> (23.04.2022)
- [25] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu, “A Survey on Deep Transfer Learning,” *arXiv:1808.01974 [cs, stat]*, Aug. 2018. [Online]. <http://arxiv.org/abs/1808.01974> (03.05.2022)
- [26] Detectron2 repository, *GitHub*. <https://github.com/facebookresearch/detectron2> (23.04.2022)
- [27] Detectron2 dokumentatsioon, “Use Custom Datasets”. <https://detectron2.readthedocs.io/en/latest/tutorials/datasets.html> (23.04.2022)
- [28] COCO - Common Objects in Context, "Detection evaluation". [Online] <https://cocodataset.org/#detection-eval> (24.04.2022)
- [29] Y. Wu *et al.*, “Demystifying Learning Rate Policies for High Accuracy Training of Deep Neural Networks,” *arXiv:1908.06477 [cs, stat]*, Oct. 2019, [Online]. <http://arxiv.org/abs/1908.06477> (11.05.2022)
- [30] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” *arXiv:1506.02640 [cs]*, May 2016. [Online]. <http://arxiv.org/abs/1506.02640> (30.04.2022)
- [31] *ultralytics/yolov5*. Ultralytics, 2022. [Online]. <https://github.com/ultralytics/yolov5> (01.05.2022)

Lisa 1 – Lihtlitsents lõputöö reprodutseerimiseks ja lõputöö üldsusele kättesaadavaks tegemiseks¹

Mina, Triin Rüütli

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose „Metsloomade tuvastamine termopiltidelt kasutades sügavaid närvivõrke“, mille juhendaja on Martin Rebane
 - 1.1. reprodutseerimiseks lõputöö säilitamise ja elektroonse avaldamise eesmärgil, sh Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;
 - 1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.
2. Olen teadlik, et käesoleva lihtlitsentsi punktis 1 nimetatud õigused jäävad alles ka autorile.
3. Kinnitan, et lihtlitsentsi andmisega ei rikuta teiste isikute intellektuaalomandi ega isikuandmete kaitse seadusest ning muudest õigusaktidest tulenevaid õigusi.

11.05.2022

¹ Lihtlitsents ei kehti juurdepääsupiirangu kehtivuse ajal vastavalt üliõpilase taotlusele lõputööle juurdepääsupiirangu kehtestamiseks, mis on allkirjastatud teaduskonna dekaani poolt, välja arvatud ülikooli õigus lõputööd reprodutseerida üksnes säilitamise eesmärgil. Kui lõputöö on loonud kaks või enam isikut oma ühise loomingu tegevusega ning lõputöö kaas- või ühisautor(id) ei ole andnud lõputööd kaitsvale üliõpilasele kindlaksmääratud tähtjaks nõusolekut lõputöö reprodutseerimiseks ja avalikustamiseks vastavalt lihtlitsentsi punktidele 1.1. ja 1.2, siis lihtlitsents nimetatud tähtaja jooksul ei kehti.

Lisa 2 – märtsikuu termoandmetel rakendatud treenimata närvivõrgu tulemused

2022. aasta märtsikuu termokaamera salvestusfailides on rakendatud üle treenimata objektituvastuse algoritmi. Väljavõte salvestusfailist on allpool.



Videofaili näide on leitav GitHub repositooriumist https://github.com/triinruutli/wild-animal-detection/blob/main/camera_without_training2.mp4.

Lisa 3 – ühe kategooriaga annoteeritud andmekoguga treenitud Mask R-CNN mudeli tulemus videol

Video asub GitHub repositooriumis.

<https://github.com/triinruutli/wild-animal-detection/blob/main/Mask%20R-CNN/video-output-animal.mkv>

Lisa 4 – mitme kategooriaga annoteeritud andmekoguga treenitud Mask R-CNN mudeli tulemus videol

Video asub GitHub repositooriumis.

https://github.com/triinruutli/wild-animal-detection/blob/main/Mask_R-CNN/video-output-deer.mkv

Lisa 5 – mitme kategooriaga annoteeritud andmekoguga treenitud YOLOv5 mudeli tulemus videol

Video asub Github repositooriumis.

https://github.com/triinruutli/wild-animal-detection/blob/main/YOLOv5/output_deers.mp4

Lisa 6 – erinevates ilmastikutingimustes Yolov5 mudeli rakendamine

Vihmastes tingimustes rakendatud Yolov5 mudeli tulemused videofailil:

<https://github.com/triinruutli/wild-animal-detection/blob/main/YOLOv5/output-rain.mp4>

Lumistes tingimustes rakendatud YOLOv5 mudeli tulemused videofailil:

<https://github.com/triinruutli/wild-animal-detection/blob/main/YOLOv5/output-snow.mp4>