TARTU ÜLIKOOL
Arvutiteaduse instituut
Informaatika õppekava

Magnus Karlson

Piltide automaatne kirjeldamine eesti keeles

Magistritöö (30 EAP)

Juhendaja: Sven Aller

Tartu 2023

Piltide automaatne kirjeldamine eesti keeles

Lühikokkuvõte:

Käesoleva magistritöö eesmärgiks on luua rakendus, mis kirjeldab etteantud pilti eesti keeles. Rakenduse loomisega soovitakse leida laste lugemaõppimist toetav lahendus. Töös uuriti arvutinägemise tehnoloogiaid ja pildi kirjeldamise võimalusi ning loodi pildi kirjeldamise rakendus. Vaadeldi kaheastmelisi ja üheastmelisi objektituvastusmudeleid ning rakendusele otsustati valida mudel üheastmeliste mudelite seast. Rakendust anti kasutajatele testimiseks, mille käigus sooviti saada tagasiside rakenduse omaduste ja objektituvastuse võimekuse kohta. Pakuti välja rakenduse edasiarendamise suundi.

Võtmesõnad:
Arvutinägemine, süvaõpe, YOLOv8, veebirakendus, ESTVision

CERCS: P176 Tehisintellekt

Automatic description of pictures in Estonian

Abstract:

The aim of this master's thesis is to create an application that describes a given image in Estonian. By creating the application, the aim is to find a solution that supports children's learning to read. In the work, computer vision technologies and image description possibilities were investigated, and an image description application was created. Two-stage and one-stage object detection models were considered, and it was decided to choose a model from one-stage models for the application. The application was given to users for testing, during which they wanted to get feedback on the application's features and object detection capabilities. Directions for further development of the application were proposed.

Keywords:

Computer vision, deep learning, YOLOv8, website application, ESTVision

CERCS: P176 Artificial intelligence

Sisukord

Sissejuhatus

Maailmas on loodud mitmeid närvivõrgustikel põhinevaid objektide tuvastuse rakendusi, millel on erinevad eesmärgid. Selliseid rakendusi kasutatakse mitmetes eluvaldkondades nagu näiteks tööstuses, kaubanduses, liikluses, spordis, põllumajanduses, meditsiinis erinevate objektide tuvastamiseks.

Arvutinägemise tehnoloogia suund on muutunud internetipõhiste rakenduste valdkonnas väga oluliseks. Olenemata sellest, kas on vaja aru saada objektide ja teksti vahelistest seostest või tuvastada nende kategooriaid, annab objekti tuvastamise tehnoloogia selleks usaldusväärset informatsiooni [1]. Erinevate rakenduste loomisega soovitakse lahendada probleeme ja abistada inimesi, aidates kiire elutempo juures säästa aega. Probleemid võivad olla erinevad, näiteks laste lugemaõppimine või vaegnägijate iseseisev hakkamasaamine. Samuti olukord, kus inimene soovib autosõidu ajal lugeda näitkes tekste, mis sisaldavad ka pilte. Sellisel juhul oleks parem seda teha kõnesünteesi abil, kus lisaks tekstile antakse häälega edasi ka piltide sisu.

Arendatud on rakendusi, mille abil soovitakse objektituvastuse kaudu anda edasi pildi sisu tekstina või kõnena, kuid need rakendused ei ole saadaval eesti keeles või ei toeta eesti keelt. Kui näiteks pilti oleks vaja kirjeldada väikelapsele tekstina, siis kõige parem on seda teha emakeeles. Seega võiks olla selline rakendus, mis kirjeldaks pilti eesti keeles. Kui laps veel ei oska lugeda, siis võiks selline rakendus toetada lapse lugemaõppimist, või vaegnägijad, kellele peab seletama, mis on pildil, ja seda võiks samuti teha eesti keeles.

Mõte see teema valida tulenes muuhulgas ühest ammusest loost, kus lapsel olid raskused lugema õppimisega ning tema vanavanemad tegid talle käsitsi kaartide süsteemi, kus ühel kaardil oli pilt ja järgmisel sõna. See oli lapse jaoks huvitav ja aitas tal lugema õppida. See oli juba aastaid tagasi ja tundus loogiline, et võiks olla ka mingi sarnane internetipõhine eestikeelne rakendus. Rakendus võiks seejuures olla laialdasema kasutusega ning sobida ka muudeks juhtudeks.

Käesolevas magistritöös soovitakse luua rakendus, mille abil peaks saama lihtsalt kirjeldada pilti pildil asuvate objektide nimetamise kaudu loomulikus eesti keeles. Eesmärgi täitmiseks uuritakse pildi kirjeldamise võimalusi ja arvutinägemise tehnoloogiaid. Vaadeldakse olemasolevaid sarnaseid rakendusi. Erinevate tuvastusmudelite käsitlemise kaudu soovitakse leida rakendusele sobilik objekti tuvastusmudel.

Töö kirjalik osa koosneb kolmest peatükist. Esimeses peatükis kirjeldatakse pildi kirjeldamise olemust, antakse ülevaade kolmest tuvastustehnoloogial põhinevast rakendusest, esitletakse autori rakendust. Teises peatükis käsitletakse objekti tuvastamist arvutinägemise kaudu, antakse ülevaade konvolutsioonilisest närvivõrgustikust, vaadeldakse objekti tuvastamise mudelite jagunemise aluseid, vaadeldakse andmestikke, käsitletakse täpsemalt mõningaid tuntumaid tuvastusmudeleid. Kolmandas peatükis käsitletakse töö autori poolt loodud objekti tuvastamise rakendust, seejuures kirjeldatakse, millisel objekti tuvastamise mudelil rakendus põhineb, millistest osadest rakendus koosneb ja kuidas rakendus töötab, käsitletakse rakenduse testimist ja testimise tulemusi kasutajate kogemuse põhjal.

1. Pildi kirjeldamine

Ashutosh Mishra jt [2] järgi on pildi kirjelduse loomist nägemistaju kaudu peetud pikka aega keerukaks väljakutseks, mis ühendab nägemise, õppimise ja keele mõistmise. Kogu sellise teabe põhjal tuleb luua asjakohane ja grammatiliselt õige kirjeldus.

Twitteri [3] järgi annab pildi kirjeldus edasi olulise informatsiooni pildi kohta tekstina. Lisaks annab kirjeldus võimaluse pildist aru saada ka inimestele, kellel esinevad nägemishäired, kes ei ole võimelised pildist aru saama või soovivad rohkem taustainfot.

Educasia [4] järgi saab pilti kirjeldada erineva täpsusastmega. Pilti võib kirjeldada lihtsalt nii , et tuvastatakse pildil olevad objektid ja loendatakse kokku, mitu sellist objekti on (vt joonis 1). Selliselt pilti kirjeldades saab tuua pildil välja ka need objektid, mis ainult pildil toimuva tegevuse üldise kirjeldamisega välja ei tuleks.
Tuvastatavateks objektideks võivad olla inimesed, loomad, sõiduvahendid, puu- ja juurviljad, mööbli- ja tarbeesemed.

Joonis 1. Kirjeldav kirjeldus: Inimesed söövad lõunat. Loendav kirjeldus: Pildil on kuus inimest.

Allika [4] järgi saab pildil asuvate objektide loendatavuse põhjal neid jagada loendatavateks ja loendamatuteks objektideks. Kui pildil asub mitteloendatav objekt, siis ei saa öelda, näiteks pildil on üks vesi või kolm lund. Objektide arvu suurust saab väljendada lisaks objektide täpsele kokkulugemisele ka määrsõnade kaudu nagu palju, vähe, suures koguses, natuke. Näiteks vaasis on üksteist tulpi, kuid saab öelda ka, et vaasis on palju tulpe, mille järgi on arusaadav, et tulpe on

rohkem kui üks. Allika järgi võib pilti kirjeldades öelda, kes või mis ja kus pildil asub, kasutades määrsõnu üleval, all, taga, ees, keskel, vasakul, paremal. Näiteks pildil keskel istub mees, temast vasakul on mees ja paremal on naine jne (vt joonis 1).

Alex Chen [5] järgi on pildi kirjeldamise puhul oluline pildil toimuva jutustamine ja seejuures mitte kõigi detailide väljatoomine pildilt, kuna siis võib kaduma minna kõige olulisem sõnum pildi kohta. Selleks, et pilti paremini kirjeldada, võib ühe võimalusena kasutada meetodit nimega objekt-tegevus-kontekst, kus objekt on peamises fookuses, tegevus kirjeldab, mis toimub vaadeldava objektiga ja kontekst kirjeldab ümbritsevat keskkonda. Selliselt pilti kirjeldades on kirjeldus objektiivne, kokkuvõtlik ja kirjeldav. Allikas väidab, et objektiivse kirjelduse korral saab inimene ka ise luua enda arvamuse, mida pilt tähendab.

1.1 Ülevaade analoogilistest rakendustest

Arvutinägemisel põhinevaid rakendusi on edukalt kasutusele võetud erinevates tööstusharudes. Järgnevalt on toodud näited kasutatavatest rakendustest [6]:

1) Tootmises töö produktiivsuse analüüsimine, visuaalne varustuse kontrollimine, kvaliteedijuhtimine, oskuste treenimine jne;
2) Tervishoius COVID-19 tuvastamine, raku klassifitseerimine, liikumise analüüsimine, maski tuvastamine, kasvajate tuvastamine, haiguse progresseerumise hindamine jne;
3) Põllumajanduses loomade jälgimine, farmi automaatika, põllukultuuride jälgimine, õitsemise tuvastamine, istanduste seire, putukate tuvastamine, automaatne saagikoristus, saagikuse hindamine jne;
4) Transpordis sõidukite klassifitseerimne, liikumiste rikkumise tuvastamine, liiklusvoogude analüüs, parkimise hõivatuse tuvastamine, automaatne numbrimärgituvastus, jalakäijate tuvastus, juhi tähelepanelikkuse tuvastamine jne;
5) Jaekaubanduses kliendi jälgimine, inimeste loendamine, varguse tuvastamine, sotsiaalne distantseerumine jne;
6) Spordis mängija asendi tuvastamine, tulemuslikkuse hindamine, palli jälgimine jne.

Järgnevalt vaatlen rakendusi Seeing AI, Google Lens ja Smartify, kuna need rakendused on minu loodavale rakendusele kõige sarnasemad. Neid rakendusi saab inimene vahetult kasutada, kus info saadakse kas siis tekstina, häälesitlusena või sarnaste piltide vastena.

1.1.1 Google Lens

Google Lens [7] järgi saab piltide kohta anda infot erinevalt kas siis tutvustava tekstina või pakkudes pildile sarnase vaste, nagu seda teeb rakendus Google Lens. Google Lens on nutitelefoni rakendus, mis on arendatud ettevõtte Google poolt. Tegemist on arvutinägemisel põhineva tehnoloogiaga, mis kasutab nutitelefoni kaamerat objektide tuvastamiseks. Allika järgi on rakenduse Google Lens abil võimalik aru saada pildist ja kasutada seda informatsiooni näiteks teksti kopeerimiseks või tõlkimiseks, taimede või loomade identifitseerimiseks, toodete leidmiseks, sarnaste piltide leidmiseks ja muudeks tegevusteks. Wondershare [8] järgi põhineb rakenduse Google Lens teksti tuvastamise funktsioon OCR (*optical character recognition*) tehnoloogial, mille abil tuvastatakse piltidel käsitsi kirjutatud või trükitud tähemärgid ja muudetakse tuvastatud märgid redigeeritavaks tekstiks.

Allika [7] järgi rakenduse tööpõhimõte seisneb selles, et rakendus võrdleb etteantud pildil leiduvaid objekte teiste piltidega ja järjestab pildid vastavalt nende sarnasusele ja asjakohasusele. Rakendus suudab mõista, mis objektiga on tegemist, ja vastavalt sellele otsida internetist asjakohaseid vasteid. Google Lens ei kirjelda pilte otseselt, vaid võrdleb pildil asuvaid objekte teiste sarnaste vastetega Google otsingumootoris ja järjestab need pildid nende sarnasuse ja asjakohasuse alusel. Sama allika järgi on seejärel võimalik kasutajal leituid vasteid lähemalt uurida ja leida vastus oma otsingule.

1.1.2 Seeing AI

Erinevalt rakendusest Google Lens edastab rakendus Seeing AI [9] tuvastatud objekti info tekstina ja häälesitlusena. Seeing AI on tehisintellektil põhinev rakendus, mis on loodud iOS operatsioonisüsteeme kasutavatele seadmetele. Seeing AI on arendatud ettevõtte Microsoft poolt. Rakendus on mõeldud nägemisraskustega inimeste abistamiseks. Seeing AI on võimalik kasutada nii telefonis kui ka tahvelarvutis, objektide tuvastamine toimub seadme kaamera kaudu. Allika järgi ei toeta rakendus eesti keelt.

Paths to Literacy [10] järgi on rakenduse Seeing AI abil võimalik täita erinevaid ülesandeid:

a) Inimeste tuvastamine ja kirjeldamine. Rakenduse abil salvestatakse inimeste näod, et hiljem oleks võimalik neid tuvastada. Rakendus kirjeldab inimeste vanust, sugu ja emotsioone;

b) Ümbritseva kirjeldamine kaamera salvestise abil;
c) Toodete tuvustamine. Triipkoodi skaneerimise kaudu on võimalik teada saada toote nimi ja tooteinfo;
d) Rahakupüüride tuvastamine;
e) Värvide kirjeldamine;
f) Lühitekstide esitamine häälesitlusena;
g) Tekstide tuvastamine ja esitamine häälesitlusena.

1.1.3 Smartify

Sarnaselt rakendusele Google Lens ei kirjelda rakendus Smartify tuvastatud objekti tekstina, vaid otsib vaste. Smartify [11] on kunstiteose tuvastamise mobiilirakendus, mis kasutab tehisintellektil põhinevat tuvastustehnoloogiat. Rakendus on saadaval operatsioonisüsteemidele iOS, Android ja veebirakendusena. Rakenduse abil saab skaneerida kunstiteose ja seejärel võrreldakse seda andmestikus olevate kunstiteostega. Kui kunstiteos on tuvastatud, siis kuvab rakendus kunstiteose nime ja selle kirjelduse. Rakendus on mõeldud nägemispuudega inimestele abistava vahendina kunstiteose vaatamiseks või selle kohta info saamiseks. Allika järgi on võimalik selle abil kunstiteose pilti ja selle juures olevat kirjeldust nutitelefonis sisse suumida ja vaadet suurendada ning kuulata kunstiteose kirjeldust häälesitlusena.

1.2 Autori rakendus

Rakenduse peamiseks ideeks on erinevate objektide pildilt tuvastamine ja pildil nende välja toomine ning tuvastatud objektide nimetamine eestikeelse tekstina. Tuvastatavateks objektideks on inimesed, loomad, sõiduvahendid, puu- ja juurviljad, mööbli- ja tarbeesemed ja rahakupüürid.

Rakendus peaks olema kättesaadav nii arvutis kui ka nutiseadmes. Positiivne kasutajakogemus toetab rakenduse jätkuvat kasutamist, millele aitavad kaasa rakenduse disain ja kasutajasõbralikkus.
Objektide tuvastamiseks hakkab rakendus kasutama kas üheastmelist või kaheastmelist objektituvastusmudelit (praegu väljatöötatud variant kasutab üheastmelist objektituvastusmudelit).

Eelnevalt vaadeldud objektituvastus tehnoloogial põhinevaid kolme rakendust, võib jagada need info edastamise osas kaheks. Ühel juhul tagastatakse kasutajale tuvastatud objekti kohta käiv info teksti ja häälesitlusena, teisel juhul tagastatakse sarnaste objektide vasted. Käesoleva töö rakendus hakkaks edastama tuvastatud objekti kohta infot tekstina.

Kuna rakendus hakkaks kuvama infot eestikeelse tekstina, erineks ta sarnastest rakendustest eesti keele kasutamise osas. Selline rakendus võiks aidata väikeseid lapsi lugema õppimisel, aga samuti rakenduse kasutajaid, kes soovivad abi erinevate objektide tuvastamisel ja selle kohta info hankimisel.

Laste lugemisoskuse tekkimiseks on erinevaid võimalusi, selline rakendus võiks lihtsalt toetada lugema õppimist ja oleks alternatiivne meetod teiste võimaluste juures. Rakenduse kasutamine toimuks vanema juuresolekul ja järelevalvel. Insider [12] järgi hakkavad lapsed sageli nutiseadmeid kasutama juba enne lugemisoskuse tekkimist, mängides nutiseadmetes lihtsaid mänge või vaadates animatsioone. Sellise võimaluse olemasolu just nutiseadmes võiks neid aga innustada tegelema millegi arendavamaga. Nicole Washington'i artikli [13] järgi on Nebraska Concordia ülikool seisukohal, et lastele tehnoloogia tutvustamine õpetab neid toime tulema erinevate operatsioonisüsteemidega ja seadmes pakutavate võimalustega.

2. Objekti tuvastamine

2.1 Objekti tuvastamine arvutinägemise kaudu

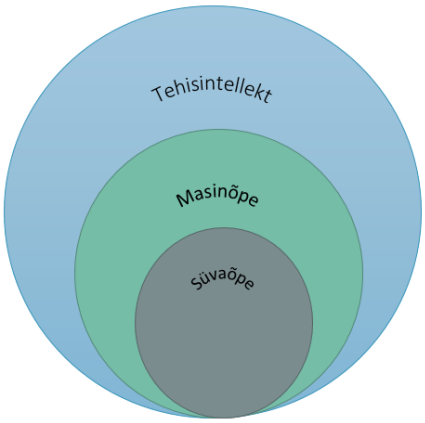
Zhengxia Zou jt [14] on kirjutanud, et objekti tuvastamine on osa arvutinägemisest (ingl *computer vision*), mille eesmärgiks on tuvastada pildilt teatud objektiklassi kuuluvate objektide esinemised. Objekti tuvastamise eesmärgiks on arendada arvutuslikke mudeleid ja tehnikaid, mis pakuvad põhiteadmisi vastamiseks arvutinägemises kerkivatele küsimustele, mis objektidega on tegemist ja kus nad asuvad. Allika järgi on objekti tuvastamisel kõige olulisemad näitajad objekti tuvastamise täpsus (sealhulgas klassifitseerimise täpsus ja asukoha määramise täpsus) ja kiirus.

Artikli "What is computer vision?" [15] järgi kuulub arvutinägemine tehisintellekti valdkonda, mis võimaldab arvutitel ja süsteemidel tuletada olulist infot piltidest, videotest ja teistest visuaalsetest sisenditest ja võtta selle teabe põhjal kasutusele meetmeid või anda soovitusi. T. Huang jt [16] on kirjutanud, et arvutinägemisel on kaks eesmärki. Bioloogia seisukohast on arvutinägemise eesmärgiks välja töötada inimese nägemissüsteemi arvutuslikke mudeleid. Allika järgi on inseneeria seisukohast arvutinägemise eesmärgiks ehitada autonoomseid süsteeme, mis suudaks täita (ja paljudel juhtudel isegi ületada) mõningaid ülesandeid, mida inimese nägemissüsteem täidab.

Artikli "What is Artificial Intelligence (AI)?" [17] järgi on tehisintellekti (AI) (ingl *artificial intelligence*) määratletud kui teadusvaldkonda, mis tegeleb arvutite ja arvutil põhinevate süsteemide ehitamisega, mis suudavad loogiliselt mõelda, õppida ja tegutseda viisil, mis tavaliselt eeldaks inimese intelligentsust või mis hõlmab andmeid, mille ulatus ületab inimeste analüüsivõime. Stuart Russell jt [18] järgi on tehisintellekti püütud defineerida nelja erineva lähenemisviisi kaudu: inimlik mõtlemine, inimlik tegutsemine, ratsionaalne mõtlemine ja ratsionaalne tegutsemine. Ratsionaalne lähenemisviis on seotud matemaatikaga ja inseneeriaga, inimkeskne lähenemisviis inimkäitumise vaatlemisega ja hüpoteesidega. Allika järgi hõlmab tehisintellekt 6 distsipliini: loomuliku keele töötlus (ingl *natural language processing*), teadmuse esitlus (ingl *knowledge representation*), automatiseeritud mõtlemine, masinõpe, arvutinägemine, robootika.

Velu Sindhu jt [19] järgi on masinõpe (ingl *machine learning*) tehisintellekti alamosa (vt joonis 2), mis õpib tuvastama mustreid etteantud andmetest, andes arvutile ülesande omandada

oskuseid, seejuures ise reegleid luues. Masinõppes kasutatakse treenimisel etteantud andmestikku (ingl *data set*) selleks, et arvuti suudaks edaspidi uute andmete omadusi ennustada. Masinõppe meetodeid kasutatakse toorandmete alusel prognoosimiseks või klassifitseerimiseks. Allika järgi suudavad sellised algoritmid tuvastada muutujate vahelisi seoseid ja interaktsioone muutujate piirides ning saab rakendada ennustuste tegemiseks, mida tuntakse ka kui regressioonimudelid.



Joonis 2. Tehisintellekt vs masinõpe vs süvaõpe [19].

Tom Mitchell jt [20] järgi tegeleb masinõpe küsimusega, kuidas mõne ülesande lahendamiseks luua arvutiprogrammi läbi kogemuse. Allika järgi kasutab masinõpe muu hulgas tehisintellekti, tõenäosuse ja statistika, arvutusliku keerukuse (ingl *computational complexity*), infoteooria, psühholoogia ja neurobioloogia, kontrolliteooria ja filosoofia ideesid.

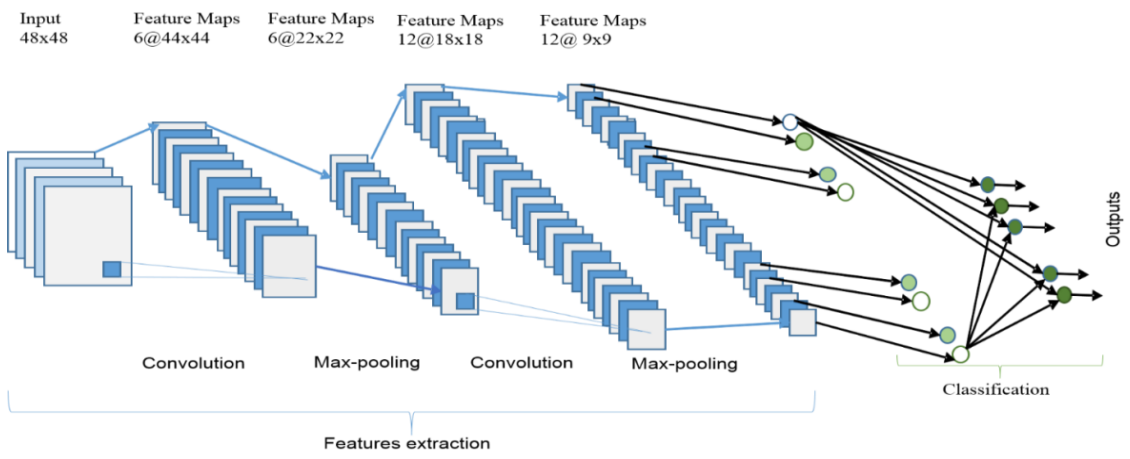
Velu Sindhu jt [19] järgi on süvaõpe masinõppe alamosa, mille algoritmid imiteerivad inimaju struktuure ja funktsioone. Zahangir Alom jt [21] järgi on süvaõpe protseduur, mis koosneb mudeli parameetrite väärtustamisest selliselt, et mudel suudaks hakata täitma spetsiifilist ülesannet. Viimatinimetatud allika järgi koosnevad tehisnärvivõrgud ANN (*artificial neural networks*) tehisneuronitest, mis proovivad jäljendada inimaju käitumist.

2.2 Konvolutsiooniline närvivõrgustik CNN

Artikli “What is a neural network?” [22] järgi on närvivõrgud, tuntud ka kui tehisnärvivõrgud ANN või simuleeritud närvivõrgud SNN (*simulated neural networks*), osa masinõppest ja süvaõppest. Närvivõrgus on tehisneuronid ehk sõlmed (ingl *nodes*) omavahel seotud kolmes kihis: sisendkihis, peitkihis ja väljundkihis. Igal sõlmel on kaal ja lävend ning see ühendub teiste sõlmedega. Kui mõne üksiku sõlme väljund ületab määratud lävendväärtust, aktiveeritakse see sõlm, mis saadab andmed võrgu järgmisele kihile.

PyCode Mates [23] järgi põhinevad konvolutsiooniline närvivõrk CNN (convolutional neural network), rekurrentne närvivõrk RNN (*recurrent neural network*) ja GAN (*generative adversarial networks*) pärilevivõrgu struktuuril. Pärilevivõrk (ingl *feed-forward network*) on tehisnärvivõrgu tüüp, mis koosneb mitmest omavahel seotud sõlmede kihist ehk mitmekihilisest pertseptronist. Need sõlmed on modelleeritud inimese aju neuronite järgi, mida kasutatakse keerukate andmete õppimiseks ja mõtestatud ennustuste tegemiseks. Allika järgi on mitmekihiline pertseptron seda tüüpi pärilevivõrk, milles andmeid edastatakse ainult ühes suunas.

Zahangir Alom jt [21] järgi koosneb konvolutsiooniline närvivõrgustik peamiselt kolme kihi kombinatsioonist (vt joonis 3). Nendeks kihtideks on konvolutsiooniline kiht, maksimaalne-ahenduskiht (ingl *max-pooling*) ja klassifitseerimise kiht. Konvolutsiooniliste kihtide ja maksimaalsete-ahenduskihtide väljundsõlmed grupeeritakse tunnuste kaardile. Iga tunnuskaardi kihid on tuletatud ühest või mitmest eelmise kihi kombinatsioonist. Kõik tasapinna sõlmed on ühendatud eelnevate kihtide väikeste alade tasapinna sõlmedega. Allika järgi kasutatakse sisendiks antud pildi tunnuste ekstraheerimiseks konvolutsioonilist kihti, kus konvolutsiooniline operatsioon eraldab pildi sisendsõlmede tunnused.



Joonis 3. Konvolutsioonilise närvivõrgustiku CNN arhitektuur [21].

Järgnevalt vaatlen Rahul Awati [24] artikli põhjal konvolutsioonilise närvivõrgustiku konvolutsioonilist kihti, ahenduskihti ja täissiduskihti.

1) **Konvolutsiooniline kiht**

   Konvolutsiooniline kiht on konvolutsioonilise närvivõrgustiku põhikomponent, kus toimub enamus arvutusi. Esimesele konvotustioonilisele kihile võib järgneda teine kiht. Konvolutsiooniprotsess hõlmab selle kihi sees asuvat filtrit, mis liigub üle pildi vastuvõtlike väljade ja kontrollib, kas pildil leidub mõni tunnus. Mitme iteratsiooni jooksul liigub tuum üle kogu pildi. Pärast iga iteratsiooni arvutatakse sisendpikslite ja filtri vahel punktkorrutis. Punktide seeria lõplikku väljundit nimetatakse tunnuskaardiks. Allika järgi teisendatakse pilt selles kihis arvväärtusteks, mis võimaldab konvolutsioonilisel närvivõrgul CNN pilti tõlgendada ja sellest olulisi tunnuseid eraldada.

2) **Ahenduskiht**

   Ahenduskiht sarnaselt konvolutsioonikihiga liigutab tuuma või filtri üle sisendpildi. Kuid erinevalt konvolutsioonikihist vähendab ahenduskiht sisendis olevate parameetrite arvu, ja lisaks põhjustab ahenduskiht ka vähest info kadu. Ahenduskiht vähendab konvolutsioonilise närvivõrgu keerukust ja parandab selle tõhusust.

3) **Täissiduskiht ehk FC (*fully connected layer*)**

Täissiduskihis toimub piltide klassifitseerimine eelmistes kihtides ekstraheeritud tunnuste põhjal. Täielikult ühendatud termin tähendab, et kõik ühe kihi sisendid või sõlmed on ühendatud järgmise kihi iga aktiveerimisüksuse või sõlmega.

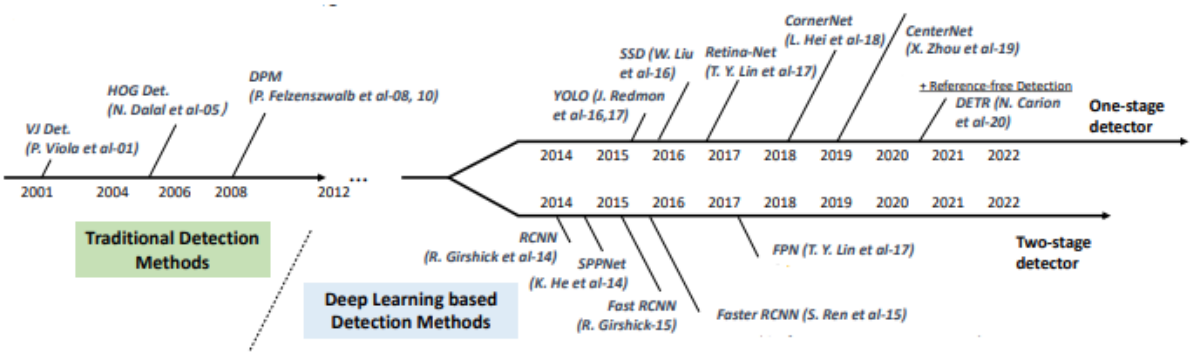
Zahangir Alom jt [21] järgi on süvaõppel (ingl *deep learning*) põhineval konvolutsioonilisel närvivõrgustikul CNN mitmeid varieeruva arhitektuuriga lähenemisviise: LeNet (1989), AlexNet (2012), ZFNet/Clarifai (2013), Network in Network (NiN) (2013), VGGNET (2014), GoogLeNet (2014), Residual Network (ResNet in 2015), Densely Connected Network (DenseNet), FractalNet (2016) (vt joonis 4).

ZFNet 2013	NiN 2013	VGGNET 2014	GoogLeNet 2014	ResNet 2015	DenseNet 2016	FractalNet 2016

Joonis 4. Tuntumad CNN arhitektuurid [21].

2.3 Tuvastusmudelite jagunemine

Zhengxia Zou jt [14] on oma töös märkinud, et üldistavalt võib viimase paarikümne aasta jooksul toimunud arengut objektituvastusraamistikes jagada kaheks perioodiks. Traditsiooniliste tuvastusmudelite periood kestis kuni 2014. aastani ja süvanärvivõrkudel põhinevate tuvastusmudelite periood algas pärast 2014. aastat (vt joonis 5).



Joonis 5. Objektituvastusraamistike jagunemine [14].

Allika [14] põhjal tuvastusmudelid jagunevad traditsioonilisteks, konvolutsioonilisel närvivõrgul baseeruvateks kaheastmelisteks ja üheastmelisteks mudeliteks:

1) **Traditsioonilised tuvastusmudelid**.

   Zhong-Qiu Zhao jt [25] on oma ülevaates kirjutanud, et traditsioonilised tuvastussüsteemid kasutavad objektide tuvastamisel visuaalsete tunnuste ekstraheerimist. Allika [14] järgi on tuntumad mudelid Viola Jones Detectors, HOG Detector ja DPM (*Deformable Part-based Model*).

2) **Konvolutsioonilisel närvivõrgul baseeruvad kaheastmelised mudelid.**

   Zhong-Qiu Zhao jt [25] järgi kaheastmelised raamistikud järgivad traditsioonilist objekti tuvastamise andmetöötlust, kus esmalt tekitatakse regiooni pakkumine objekti sisaldavatele aladele, seejärel klassifitseeritakse ja kalibreeritakse neid. Kaheastmelistel mudelitel on suur täpsus, kuid halb reaalajas (ingl *real-time*) sooritus. Allika [14] järgi tuntumad mudelid on R-CNN, SPPNet, FAST R-CNN, Faster R-CNN, Feature Pyramid Networks (FPN).

3) **Konvolutsioonilisel närvivõrgul baseeruvad üheastmelised mudelid.**

   Zhong-Qiu Zhao jt [25] ülevaate järgi muudavad üheastmelised raamistikud objekti tuvastamise regressiooni- või klassifikatsioonimudeliks, kus adopteeruv raamistik leiab tulemuse otse läbi asukoha ja klassikuuluvuse määratlemise. Allika [14] järgi tuntumad mudelid on YOLO, SSD (Single Shot MultiBox Detector), RetinaNet, CornerNet, CenterNet, DETR.

Rakendusele mudeli valimiseks vaatlesin mudelite jagunemist läbi ajatelje. Allika [14] järgi on traditsioonilised tuvastusmudelid olnud enne konvolutsioonilise närvivõrgustiku kasutuselevõtmist valdkonna suunanäitajad. Sellised mudelid põhinesid inimeste poolt etteantud tunnustel (ingl *handcrafted features*). Kuna puudusid efektiivsed meetodid pildilt tunnuste eraldamiseks, siis tuli inimestel ise luua keerulisi meetodeid pildilt tunnuste eraldamiseks ja samuti meetmeid, mis muudaksid tunnuste eraldamise protsessi tõhusamaks. Konvolutsioonilise närvivõrgustiku loomisega algas tuvastusmudelite kiire areng. Tekkisid kaheastmelised ja üheastmelised mudelid. Kaheastmelistel mudelitel on suur täpsus, kuid halb reaalajas sooritus. Need mudelid on keerulised, kuna järgivad traditsioonilist objekti tuvastamise andmetöötlust, kus esmalt tekitatakse regioonipakkumine objekti sisaldavatele aladele ja seejärel klassifitseeritakse ja kalibreeritakse neid. Üheastmelised tuvastusmudelid suudavad objekte

tuvastada üheetapiliselt, muutes objekti tuvastamise regressiooni- või klassifikatsioonimudeliks, kohanev piirikast leiab tulemuse otse läbi asukoha ja klassikuuluvuse määratlemise. Üheastmeliste tuvastusmudelite probleemiks on tihedalt koos olevate ja väikeste objektide tuvastamine. Allika järgi sobivad üheastmelised mudelid nutiseadmetes kasutamiseks, kuna nendel mudelitel on hea reaalajas soorituskiirus ja neid on võimalik kergesti kasutusele võtta.

Rakendusele mudeli valimisel otsustasin vaadelda nii kaheastmeliste kui ka üheastmeliste tuvastusmudelite esindajaid, mida käsitlen lähemalt alapeatükis 2.6.

2.4. Andmestikud

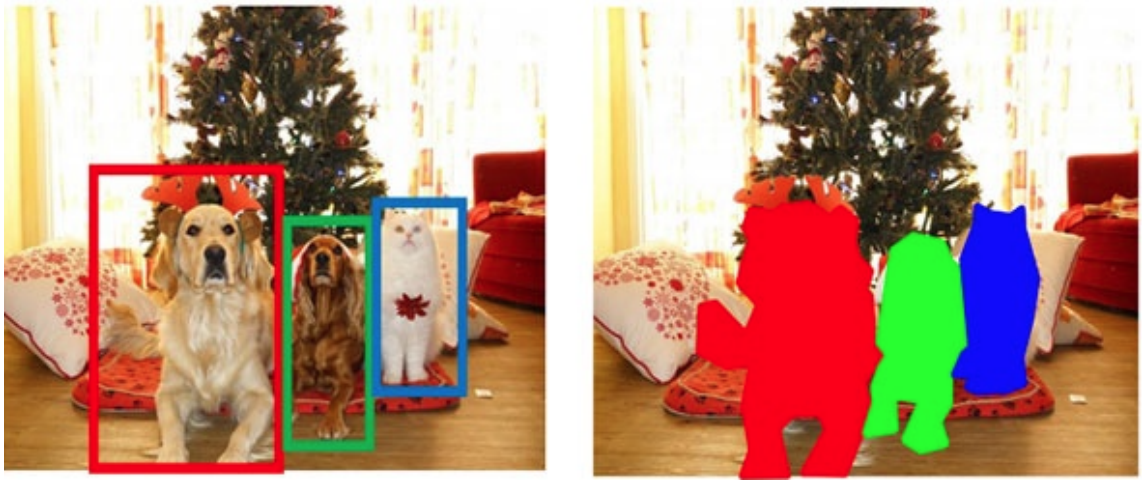
Shehmir Javaid [26] järgi on masinõppe andmestik andmete kogum, mida kasutatakse näiteks mudeli treenimiseks. Andmestiku abil on võimalik õpetada masinõppe algoritmi ennustusi tegema. Levinumad andmetüübid on tekstiandmed, pildiandmed, heliandmed, videoandmed ja arvandmed. Andmed on tavaliselt kõigepealt märgendatud, et algoritm saaks aru, milline peab olema ennustatav tulemus. Sobiliku andmestiku ettevalmistamine ja valimine on masinõppe mudeli treenimise üks olulisemaid samme. Allika järgi jagunevad masinõppe andmestikud treeningandmeteks, valideerimisandmeteks ja testandmeteks.

Kent Gauen jt [27] järgi vajavad keerukad tuvastusmudelid piltidelt ja videotelt objektide tuvastamiseks suurel hulgal andmeid. Visuaalsete andmete töötlemine on toonud kiire tehnoloogilise arengu arvutinägemises. Arvutinägemise arengusse panustamisel on lisaks muudele faktoritele oluline roll ka märgendatud andmetel. Paljud andmestikud nagu näiteks ILSVRC (*ImageNet Large Scale Visual Recognition Challenge*), COCO (*Common Objects in Context*), PASCAL VOC (*Pattern Analysis Statistical Modelling and Computional Learning Analysis Statistical Modelling and Computional Learning*) on loodud internetist piltide otsimise ja allalaadimise teel. Lisaks kogutakse pilte auto pardakaamera abil jäädvustades, sellist meetodit kasutavad andmestikud KITTI (*Toyota Technological Institute at Chicago Object Detections*) ja Caltech (*Caltech Pedestrian Datasets*). Andmestike loomisel on kõige keerulisem osa andmete märgendamine. Allika järgi ei saa andmestike märgendamist automatiseerida masinaõppe abil, kuna neid andmestikke kasutatakse masinõppe mudeli treenimiseks ning seetõttu tuleb märgendamine teha inimeste poolt manuaalselt. Allika [28] järgi määratletakse esmalt

pildikogumid kategooriate järgi, seejärel kogutakse internetist pildid eri kategooriaid esindavatesse pildikogumitesse ja lõpuks kogutud pildid märgendatakse.

Alljärgnevalt on välja toodud tuntumad andmestikud:

a) PASCAL VOC. Mark Everingham jt [29] järgi on PASCAL VOC nähtavate objektide andmestik. PASCAL VOC andmestik on esitatud aastate lõikes 2005-2012. PASCAL VOC andmestik täidab kahte eesmärki. Üheks eesmärgiks on märgendatud piltide andmestiku ja standardiseeritud hindamistarkvara kõigile kättesaadavaks tegemine ja teiseks eesmärgiks on iga-aastaste võistluste ja töötubade korraldamine. PASCAL VOC andmestikul on viis väljakutset: klassifitseerimine, tuvastamine (vt joonis 6), segmenteerimine (vt joonis 6), tegevuste klassifitseerimine ja inimeste paigutus. Andmestik Pascali VOC koosneb 20 objektikategooriast. Allika järgi on andmesiku PASCAL VOC lõikeühiku *IoU* (*intersection over union*) lävendiks 0.5.



Joonis 6. Objekti tuvastamine ja segmenteerimine [30].

b) ImageNet [31] järgi on ILSVRC suuremahuline objektituvastuse ja pildi klassifitseerimise võistlus. Võistlus aitab võrrelda tuvastusmeetodite edusamme laiema objektivaliku hulgas. Lisaks on võimalik võrrelda arvutinägemise edusamme ka suure pildikogu indekseerimisel ning piltide otsimisel ja märgendamisel. Bulent Siyah [32] järgi ILSVRC andmestik kasutab ImageNet alamhulka, kus on ligikaudu 1000 pilti 1000 objekti kategoorias ehk treenimiseks on ligikaudu 1.2 miljonit pilti, valideerimiseks 50

000 pilti ja testimiseks 150 000 pilti. Allika [33] järgi võiks olla ILSVRC andmestikus teatud juhul olla väikeste objektide kõrvalkallete vältimiseks lõikeühiku *IoU* lävendiks 0.25. Sik-Ho Tsang [34] järgi määrati ImageNet andmestikus NMS (ingl *non-maximum suppression*) lõikeühiku *IoU* lävendiks 0.3, kuid empiiriliselt on leitud, et 0.4 lävend on parem.

c) COCO [35] järgi on MS COCO andmestik, mille abil on võimalik teha objektide tuvastamist, segmenteerimist ja pildi kirjeldust. Pildikogu koosneb ligikaudu 330 tuhandest pildist, millel on objektid manuaalselt märgendatud ja määratletud asukohad välja toodud. Andmestikus on üle 80 objekti kategooria. Allika järgi on andmestikust COCO saanud üks populaarseimaid andmestikke, mida kasutatakse laialdaselt objektituvastuse treenimiseks ja valideerimiseks. Allika [36] järgi on COCO andmestikus lõikeühiku *IoU* lävendid vahemikus 0.5–0.95, sammuga 0.05 ja seda hinnatakse eraldi väikeste, keskmiste ja suurte objektide puhul.

d) Roboflow100. Floriana Ciaglia jt [37] järgi on Roboflow100 uus objektide tuvastamise andmestik, kus keskendutakse ühe andmestiku ühe mõõdiku optimeerimisele. Roboflow100 on kasutamise juurdepääsuga avalik andmestik erinevate märgendusformaatidega. Andmete eeltöötlemine on seotud pildi kõrgus-laiussuhte, suuruse ja kontrastsuse määratlemisega ning andmete märgendamine. Allika järgi Roboflow100 koosneb 100-st objektituvastuse andmestikust, mille on spetsiaalselt koostanud Roboflow kasutajad.

Andmestikud PASCAL VOC, ILSVRC, MS COCO ja Roboflow100 kasutavad objektituvastusmudelite soorituse täpsuse hindamisel näitajaid *AP* (*average precision*) ja *mAP* (*mean average precision*). Erinevates andmestikes näitajate väärtused varieeruvad erinevate hindamiskriteeriumite ja kasutatavate andmete tõttu, seetõttu ei saa otseselt võrrelda ühes andmestikus saadud näitajate tulemusi teises andmestikus saadud samade näitajate tulemustega.

2.5 Täpsusmõõdikud

Käesolevas alapeatükis on käsitletud mõõdikuid *AP* ja *mAP*. Kiprono Elijah Koech [38] allika põhjal on *AP* ja *mAP* kõige populaarsemad mõõdikud, mida kasutatakse objektide tuvastusmudelite, näiteks R-CNN, Faster R-CNN, YOLO, SSD, hindamiseks. Samuti on need

mõõdikud kasutusel objektituvastuse pakkumiste hindamisel andmekogudes COCO ja PASCAL VOC.

Vijay Dubey [39] järgi hinnatakse objektituvastamise algetapil kõigepealt, kas tuvastamine on õige. Selleks kasutatakse abistavat näitajat - lõikeühikut *IoU*. Lõikeühik *IoU* on Jaccardi indeksil põhinev mõõt, mida kasutatakse valimikomplektide sarnasuse ja mitmekesisuse mõõtmiseks. Tuvastusmudelite puhul hindab lõikeühik *IoU* ennustuskasti-tõekasti ühisosa (ingl *area of overlap*) ja ennustuskasti-tõekasti ühendi (ingl *area of union*) suhet, mida saab leida valemiga (1), kus *gt* tähistab tõekasti väärtust ja *pd* tähistab ennustuse väärtust [29].

.

$$IoU = \frac{area\,(gt \cap pd)}{area\,(gt \cup pd)} \qquad (1)$$

Kiprono Elijah Koech [38] järgi on lõikeühiku *IoU* kasutamiseks vaja lävendit (näiteks α), mille kaudu saab kindlaks teha, kas tuvastamine on õige. Lõikeühiku *IoU* suurus võib varieeruda vahemikus 0 ja 1, kus 0 näitab, et kattuvus puudub, ja 1 näitab tõeväärtuse *gt* ja ennustuse *gt* täielikku kattuvust.

Alljärgnevalt on välja toodud lõikeühiku *IoU* ja lävendi *α* suhestumine ennustuse ja tõeväärtuse kaudu.

1) Kui lõikeväärtus $IoU \geq \alpha$, siis on tuvastus tõepositiivne TP (ingl *true positive*). Mis tähendab, et tuvastamine on täpne.
2) Kui lõikeväärtus $IoU < \alpha$, siis on tuvastus valepositiivne FP (ingl *false positive*). Mis tähendab, et tuvastamine on vale.
3) Kui lõikeväärtus $IoU = 0$, siis on tuvastus valenegatiivne FN (ingl *false negativ*e). Mis tähendab, et ei tuvastatud.
4) Näitajat tõenegatiivne TN (ingl *true negative*) ei kasutata.

Näiteks kui lõikeühiku *IoU* lävendi väärtus on $\alpha = 0.5$, siis kõik *IoU* väärtused, mis on lävendiga võrdsed või suuremad, on tõepositiivsed tuvastamised ja kõik väärtused, mis on lävendist allpool, on ebatäpsed tuvastamised. Kui lõikeühiku *IoU* väärtus on 0, siis tuvastamist ei toimunud.

Allika [38] järgi näitab täpsus (ingl *precision*), kui palju sobivatest ehk tõeseks hinnatutest ka tegelikult sobivad ehk on tõesed, täpsuse leidmiseks kasutatakse valemit (2). Saagis (ingl *recall*) näitab, kui palju kõigist sobivatest elementidest suudeti leida sobivaid elemente, saagise leidmiseks kasutatakse valemit (3).

$$P = \frac{TP}{TP + FP} = \frac{TP}{kõik\ tuvastused} \quad (2)$$

$$R = \frac{TP}{TP+FN} = \frac{TP}{kõik\ sobivad\ elemendid} \quad (3)$$

Allika [38] järgi täpsuse ja saagise kõver (ingl *precision recall curve)* näitab tuvastusmudeli sooritust graafiliselt, kus *y*-teljel asuvad täpsuse näidud ja *x*-teljel saagise näidud. Täpsusnäitaja *AP* on tõepositiivsete juhtude ja mudeli tehtud positiivsete ennustuste koguarvu suhe. See mõõdab, kui hästi mudel väldib valepositiivseid tulemusi. Täpsusnäitaja *AP* saab arvutada valemiga (4).

$$AP = \int_0^1 p(r)dr \quad (4)$$

Näitaja *AP* väärtust arvestatakse iga klassi kohta üksikult. See tähendab, et *AP* väärtusi on sama palju kui objektiklasse. Näitaja *mAP* on näitaja *AP* kõigi klasside väärtuste keskmine, mis arvutatakse valemiga (5).

$$mAP = \frac{1}{n}\sum_{i=1}^{n} AP_i \quad (5)$$

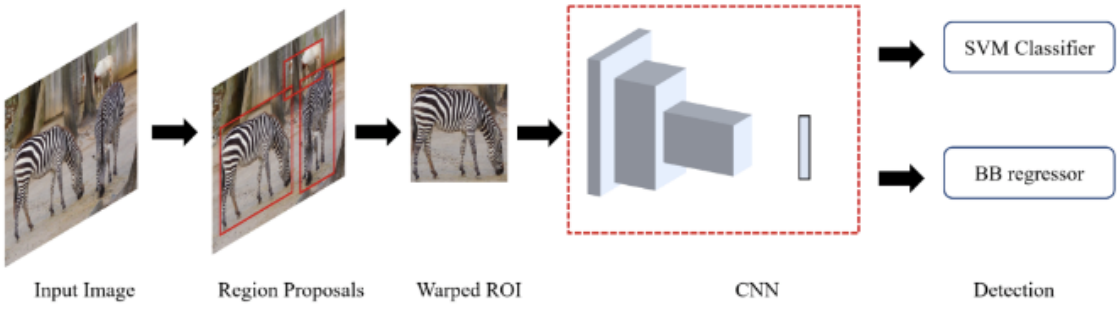
2.6 Kaheastmeline või üheastmeline mudel

Alljärgnevates alapeatükkides vaadeldakse objektituvastusmudeleid R-CNN, Faster R-CNN, YOLO, YOLOv8 ja SSD. Mudelite tutvustuste juures on välja toodud objektituvastusmudelite mõõtmistulemused andmestikes PASCAL VOC, ILSVR2013, COCO ja Roboflow100. Töö kirjutamise ajal mudeli YOLOv8 ametlikku raportit ei olnud välja antud.

2.6.1 R-CNN mudel

Käesolevas alapeatükis vaadeldakse mudelit R-CNN Ross Girshick jt [40] töö põhjal. R-CNN on objekti tuvastamise kaheastmeline mudel, kus toimub eraldi asukoha määratlemine ja klassifitseerimine (vt joonis 8). Mudel R-CNN põhineb konvolutsioonilisel närvivõrgustikul CNN. Asukoha pakkumine (ingl *region proposal*) on R-CNN mudeli põhikontseptsiooniks. Pildil määratakse objektide asukohad läbi regiooni pakkumise.

Objekti tuvastamiseks kõigepealt mudel ekstraheerib pildil regiooni pakkumised, milleks kasutatakse valikulise otsimise algoritmi. Valikulise otsingu algoritm genereerib pildist alamsegmente, mis võivad kuuluda ühele objektile – värvi, tekstuuri, suuruse ja kuju alusel – ning kombineerides korduvalt sarnaseid piirkondi objektide moodustamiseks. Seejärel muudetakse ekstraheeritud väljalõigete suurust ja suunatakse need läbi närvivõrgu. Lõpuks määratakse närvivõrgus väljalõikele kategooria $C + 1$. Allika järgi ennustatakse täiendavalt väjalõike suuruse määratlemiseks koordinaatide X ja Y väärtused.



Joonis 8. R-CNN mudeli arhitektuur [41].

Objektide asukoha määratlemiseks kasutatakse valikulise otsimise algoritmi, mis rühmitab regiooni pakkumisi nende pikslite intensiivsuse alusel, st läbi sarnaste pikslite hierarhilise rühmitamise moodustatakse pikslite grupid. Mudel võib ekstraheerida umbes 2000 asukoha pakkumist. Ekstraheeritud asukohaettepanekud märgendatakse läbi treenimise. Objekti tuvastamisel piirikastid (ingl *bounding-box*) märgendatakse koos nende klassikuuluvuse pakkumistega läbi lõikeühiku *IoU* väärtuste. Allika järgi peab lõikeühiku *IoU* väärtus olema

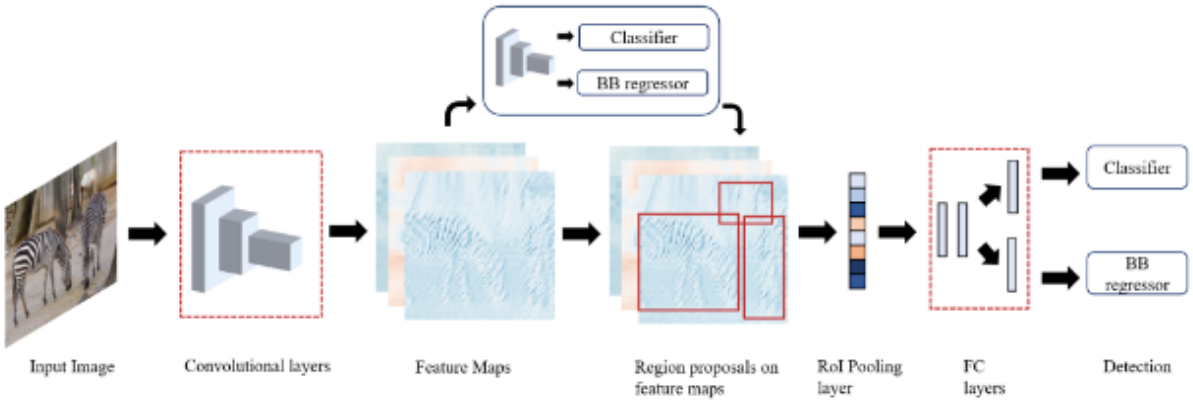
suurem või võrdne 0.5-ga, kui *IoU* väärtus on väiksem kui 0.5, siis määratletakse seda pildi taustana ja ei võeta arvesse.

Ross Girshick jt [40] järgi saavutas R-CNN mudel testimisel PASCAL VOC 2010 andmestikul *mAP* tulemuseks 53.7% ja ILSVRC2013 andmestikul *mAP* tulemuseks 31.4%.

2.6.2 Faster R-CNN mudel

Alljärgnevalt vaadeldakse Faster R-CNN mudelit Shaoqing Ren jt töö [42] põhjal. Faster R-CNN mudel on varasema mudeli Fast R-CNN laiendus. Faster R-CNN koosneb kahest moodulist - regioonipakkumiste võrgust RPN (*region proposal network*) ja Fast R-CNN mudeli objekti tuvastamise detektorist (vt joonis 9). Esimeseks sammuks on sisendpildil tunnuste kaardistamine (ingl *feature mapping*), selleks on kasutatud konvolutsioonilist närvivõrku. Seejärel saadetakse pildi tunnuste kaart regioonipakkumiste võrgule RPN. RPN on täielikult konvolutsiooniline võrk, mis genereerib erineva ulatuse ja kuvasuhtega kastikesi, kust võib tuvastada objekti. Iga selline kastikene kannab objekti olemasolu väärtust, mis tähistab seda, kui palju kuulub kasti sisu objektiklasside hulka. Valitud pakkumised kaardistatakse seejärel eelmisest CNN-i kihist saadud tunnuste kaardiga ja saadetakse lõpuks RoI pooling (*region of interest pooling*) operatsiooni läbimisel klassifitseerimisele.

Mudeli Faster R-CNN treenimisel on regioonietteppanekute võrku RPN eeltreenitud ImageNeti andmestikus ja peenhäälestatud PASCAL VOC andmestikus. Faster R-CNN mudeli tuvastamistäpsus on varasemate mudelitega võrreldes paranenud rohkem kui 3%.



Joonis 9. Faster R-CNN mudeli arhitektuur [41].

Shaoqing Ren jt [42] järgi saavutas Faster R-CNN mudel testimisel PASCAL VOC 2007 andmestikul *mAP* tulemuseks 69.9% ja PASCAL VOC 2012 andmestikul *mAP* tulemuseks 67%, COCO treeningandmeid kasutades PASCAL VOC 2007 testis oli *mAP* tulemus 76.1% ja PASCAL VOC 2012 testis oli *mAP* tulemus 73%.

2.6.3 YOLO mudel

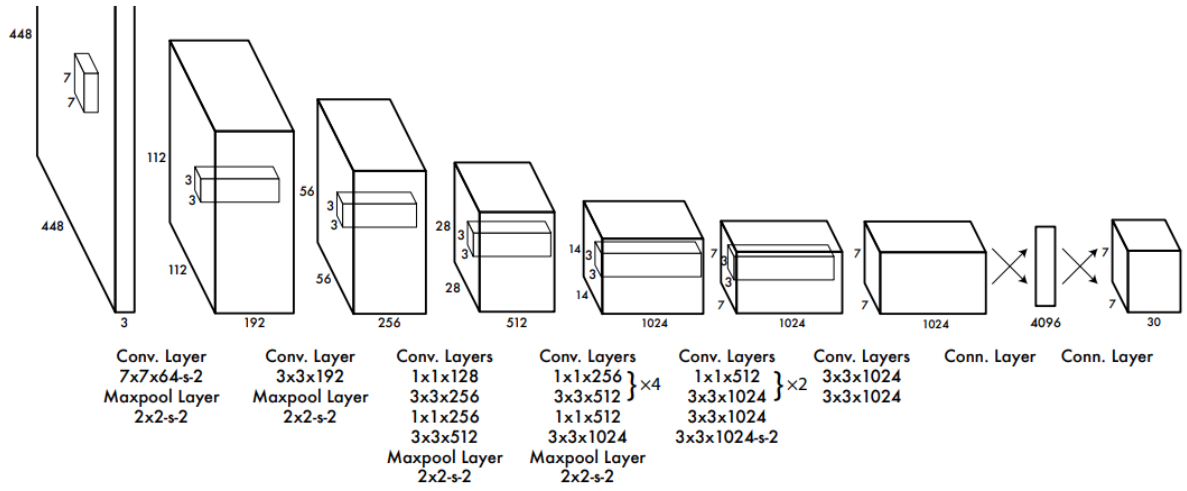
Käesolevas alapeatükis vaadeldakse YOLO mudelit Joseph Redmon jt töö [43] põhjal. YOLO (*You Only Look Once)* on piltidel ja videotel asuvate objektide tuvastamise ja määratlemise algoritm. Objekti tuvastamise käigus identifitseeritakse ühe või mitme objekti täpne asukoht objektide nelinurkse raamistamise kaudu.

Autorite sõnul on nad kujundanud objekti tuvastamise ümber üheks regressioonimudeliks, kus piltide pikslitest saavad piirikastide koordinaadid ja tõenäolised klassid. YOLO mudel ennustab, kes või mis on pildil ja kus nad asuvad. Konvolutsiooniline närvivõrk CNN ennustab samaaegselt piirikastide sisu ja klassi kuuluvust.
Objekti tuvastamine toimub YOLO meetodil eraldiseisvate komponentide ühendamise kaudu üheks närvivõrguks. Ennustades piirikastide sisu, kasutab lahendus närvivõrgu funktsioone terve pildi ulatuses. Pildil asuvate raamistatud objektide tuvastamine toimub samaaegselt. YOLO disain võimaldab teha reaalajas ennustusi, millel on kõrge keskmine ennustuse täpsus.

Sisestatud pilt jagatakse *S x S* ruudustikuks (ingl *grid*). Kui objekti keskpunkt langeb ruudustiku lahtri keskele, siis see lahter vastutab objekti tuvastamise eest. Iga lahter ennustab *B* piirikaste ja ja nende usaldusmäära. Usaldusmäär näitab, kui kindel on mudel ennustamisel, kas lahter sisaldab objekti ja kas piirikast on täpne. Usaldust defineeritakse kui *Pr (Objekt)* * *IoU* (5). Kui lahtris ei ole objekti, siis on usaldusmäär 0.
Iga piirikastil on viis ennustuse näitajat: *x, y, w, h* ja usaldusmäär. Koordinaadid *x* ja *y* tähistavad piirikasti keskosa, mis vastab ruudustiku lahtri piiridele. Kõrgused ja laiused on kogu pildi suhtes ennustatavad. Usaldusmäär näitab ennustuskasti ja tõekasti vahelist seost. Lisaks ennustab ruudustiku iga lahter tingimuslikult objektiklassi *C* tõenäosust*, Pr (Klass|Objekt)*. Need

tõenäosused on tingimuslikult antud objekti sisaldavatele ruudustiku lahtritele. Olenemata *B* piirikastide arvust, ruudustiku lahtri kohta ainult ennustatakse klassi tõenäolisuseid.

YOLO mudelit rakendatakse konvolutsioonilise närvivõrguna. Võrgu esimesed konvolutsioonikihid ekstraheerivad pildi tunnused, täissiduskihid ennustavad väljundi tõenäosusi ja koordinaate. Mudeli arhitektuuri väljatöötamisel on saadud inspiratsiooni GoogLeNet pildi klassifitseerimise mudelist. YOLO mudeli närvivõrgul on 24 konvolutsioonikihti, millele järgneb 2 täissidusat kihti (vt joonis 10).



Joonis 10. YOLO mudeli arhitektuur [43].

Mudeli väljundi optimeerimiseks kasutatakse summa-ruudu viga (ingl *sum-squares error*), mida on lihtne optimeerida, kuid mis ei vasta eesmärgile maksimeerida keskmist täpsust.

Autorid on välja toonud, et YOLO mudelil on tugevad piirangud piirikastide ennustamisel, kuna ruudustiku üks lahter ennustab ainult kahte piirikasti ja selles saab olla ainult üks klass. Selline ruumiline kitsendus seab piirid ennustatavate objektide arvule. Mudelil tekivad raskused väikeste objektide tuvastamisel grupis, näiteks linnuparved. YOLO mudelil on raskuseid uute ebatavaliste kuvasuhete ja konfiguratsioonidega objektide üldistamisel, kuna mudel õpib otse andmete põhjal. Piirikastide ennustamisel kasutatakse suhteliselt üldistavaid tunnuseid (ingl *coarse features*), kuna mudeli arhitektuuril on mitu maksimaalset-ahenduskihti.

Mudeli puhul kaofunktsiooni kasutamisega soovitakse kõrvaldada vead nii suurtes kui ka väikestes piirikastides. Üldjuhul väike viga väikeses kastis avaldab suuremat mõju ristlõikeühikule *IoU*, kui väike viga suures kastis. Mudeli suurimaks veaallikaks on ebatäpsed piirkonna määramised (ingl *localizations*).

Joseph Redmon jt [43] järgi saavutas YOLO mudel testimisel PASCAL VOC 2012 andmestikul *mAP* tulemuseks 57.9%, COCO treeningandmetega PASCAL VOC 2007 testis oli *mAP* tulemus 76.1% ja PASCAL VOC 2012 andmestikul oli *mAP* tulemus 73%. Allika järgi oli YOLO mudeli *mAP* 57.9% tulemus madalam võrreldes R-CNN mudeli *mAP* 70.7% tulemusega, kus põhjusena on välja toodud mudeli raskused tuvastada väikseid objekte.

Awesome-yolo [44] järgi võeti YOLO mudel kasutusele 2015. aastal. Mudelit on mitmeid kordi edasi arendatud erinevates versioonides: YOLO, YOLO 9000 (v2), YOLOv3, YOLOv4, YOLOR, YOLOX, YOLOv5, YOLOv6, YOLOv7, YOLOv8. Erinevalt varasematest versioonidest on YOLOv6 versioon välja töötatud suure jõudlusega tööstulikele rakendustele. Allika järgi mudeli viimane edasiarendus on YOLOv8, mis võeti kasutusele 2023. aastal.

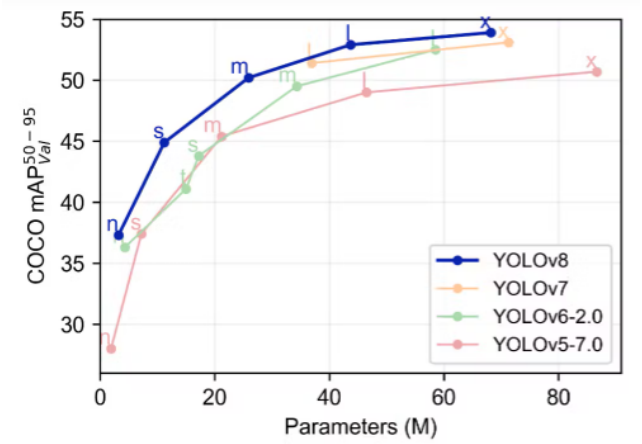
2.6.4 YOLOv8 mudel

Käesolevas alapeatükis vaadeldakse YOLOv8 mudelit peamiselt Jacob Solawetz artikli [45] põhjal. YOLOv8 mudel on uusim YOLO mudelite seas. Mudeli abil saab piltidelt objekte tuvastada, segmenteerida ja klassifitseerida. YOLOv8 on YOLOv5 edasiarendus, millel on täiustatud mudeli arhitektuuri ja lisaks on arvestatud arendajatega, et arendajatel oleks lihtsam YOLOv8 mudelit kasutada. Lisaks on YOLOv8 mudel nüüd kättesaadav PIP (*package installer for Python*) paketina. PIP kujutab endast Pythoni paketi paigaldustööriista, mille abil saab koodile juurde lisada programmeerimiskeele Python pakette [46].

Mudeli suurimaks muudatuseks on ankruvaba tuvastamine, mis kujutab endast otse objekti keskpunkti leidmist, ilma eelnevalt defineeritud ankrukastist nihet leidmata. Ankrukastid on eelnevalt defineeritud piirikastid, millel on kindel pikkus ja laius ning neid kasutatakse objektiklassi tuvastamiseks, millel on sobiv mõõtkava ja kuvasuhe [47]. Ankruvaba tuvastamine vähendab ennustuste arvu piirikastis, mis kiirendab NMS järeltöötlusetappi, mille käigus eemaldatakse duplikaat-objektituvastused ja jäetakse alles kõige tõenäolisemad objektiklassid.

Mudeli treenimisel kasutatakse mosaiigile sarnast andmete suurendamist, mis kujutab endast nelja erineva treeningpildi kokkuliitmist üheks tervikuks erinevates osakaaludes. See aitab mudelil õppida objekte tuvastama erinevates asukohtades, osalise ärakaetusega ja erinevate ümbritsevate pikslite hulgast. Kuid sellist andmete suurendamist ei kasutata terve treeningprotsessi vältel, kuna katsetest on selgunud, et mudeli objektituvastuse võimekus läheb halvemaks.

Nikolaj Buhl artikli [47] järgi on YOLOv8 mudelil on erinevate treenitavate parameetrite arvuga mudeleid, nendeks mudeliteks on YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l ja YOLOv8x. YOLO seeria mudelite *mAP* ja parameetrite arvu vahelised seosed on toodud välja graafikul (vt joonis 10), kus mudeleid on võrreldud COCO andmestikul. Lisaks on graafikul välja toodud YOLO seeria varasemate mudelite parameetrite ja *mAP* väärtuste suhe ning graafikult on näha, et YOLOv8 mudelil on kõrgem *mAP* väärtus madalama parameetrite arvu korral.



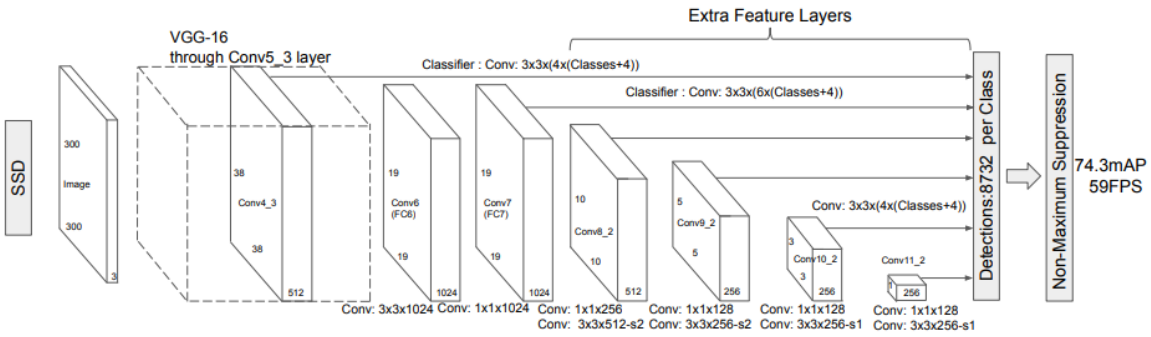
Joonis 10. Mudelite *mAP* väärtuse ja´parameetrite vaheline seos [47].

2.6.5 SSD mudel

Käesolevas alapeatükis vaadeldakse SSD mudelit Wei Liu jt [48] töö põhjal. SSD (*Single Shot Multibox Detector*) on ühtsel närvivõrgul põhinev piltide üheastmeline objektituvastusmudel. SSD mudel põhineb pärilevi konvolutsioonilisel närvivõrgustikul, mis genereerib fikseeritud suurusega piirikastid. Lisaks on igal piirikastil objektiklassi tõenäosused, kui suure tõenäosusega

võib objektiklass leiduda piirikastis. Mudelis kasutatav närvivõrk koosneb mitmest osast, närvivõrgu alguses kasutatakse VGG-16 võrgustikku (vt joonis 11), mida kutsutakse ka alusnärvivõrguks. Alusnärvivõrgule järgnevad abistavad närvivõrgustiku kihid. Mitmemõõtmeliste tunnuskaartide tuvastamiseks lisatakse alusnärvivõrgu lõppu konvolutsioonilised tunnuskihid. Lisatud tunnuskihid lähevad järk-järgult väiksemaks, et ennustusi teha mitmel skaalal. Iga tunnuskiht saab genereerida konvolutsiooniliste filtrite abil kindla hulga objektituvastuse ennustusi. Konvolutsioonilise filtri abil saab pildilt tunnuseid eraldada, tunnusteks võivad olla nurgad, servad või tekstuurid. Sellele järgneb NMS samm, mille käigus eemaldatakse duplikaat-objektituvastused ja jäetakse alles kõige tõenäolisemad objektiklassid.

SSD mudeli treenimine erineb teiste mudelite treenimisest sellepoolest, et treeningandmete väärtused tuleb määrata objektituvastusmudeli väljundi spetsiifilistele väljunditele. Allika järgi toimub seejärel mudeli treenimine *end-to-end* meetodi kaudu, mis kujutab endast terve mudeli treenimist sisendist kuni väljundini. Lisaks sisaldab mudeli treenimine hulga algsete piirikastide valimist erinevas suuruses ja andmete tehislikku suurendamist, mille käigus pilte modifitseeritakse.



Joonis 11. SSD mudeli arhitektuur [48].

Wei Liu jt [48] järgi saavutas SSD mudel testimisel PASCAL VOC 2007 andmestikul *mAP* tulemuseks 68%, PASCAL VOC 2012 andmestikul oli *mAP* tulemus 72.4% ja ILSVRC2013 andmestikul oli *mAP* tulemus 43.4%.

3. Eestikeelne rakendus

Käesolevas peaktükis vaadeldakse rakenduse komponente, milleks on kasutajaliides, taustaprogramm ja veebiserver. Antakse ülevaade rakenduse komponentidest, tuuakse välja erinevad valikuvariandid ja milline komponent on valitud. Lisaks antakse ülevaade rakenduse testimisest ja selle tulemustest.

3.1 Kasutajaliides

Fred Churchville [49] järgi on kasutajaliides tarkvara või süsteemi visuaalne ja interaktiivne osa, mis võimaldab kasutajatel süsteemiga suhelda ja konkreetseid ülesandeid täita. Kasutajaliidese eesmärk on pakkuda kasutajale sujuv ja intuitiivne kasutajakogemus. Mõned võimalikud variandid kasutajaliideseks on veebiliides, töölauarakendus ja mobiilirakendus.

Rakenduse jaoks otsustasin valida veebiliidese, kuna see tagab ligipääsu rakendusele nii nutitelefoni kui ka arvuti kaudu läbi veebibrauseri.

3.1.1 Kasutajaliidese kujundus

Kasutajaliidese kujundus omab olulist rolli, kuna sellega puutub kasutaja koheselt kokku, kui kasutaja veebisaiti külastab. Kujundus vastutab toote välimuse, interaktiivsuse, kasutatavuse ja üleüldise tunde eest. Sellest sõltub, kas kasutajal on positiivne kogemus või mitte. Tomasz Bąk [50] järgi on hea kasutajaliidese kujunduse prioriteetideks lihtsus ja kasutatavus, et luua mõjuv visuaalne keel toote kasutajatele. Allika kohaselt on kujunduse puhul mõningad trendid, mis aitavad kujunduse loomisele kaasa:

a) Lame kujundus, mis on populaarne lähenemine veebisaidi kujundamisel, mille eesmärgiks on luua võimalikult lihtne ja intuitiivne kasutajakogemus. Peamine fookus on minimalismil ja kasutatavusel, mitte säravatel visuaalsetel efektidel. Kasutab minimalistlikku värvipalleti ja kujundeid, et luua esteetiliselt meeldiv kujundus.
b) Reageeriv kujundus - tegemist on lahendusega, kus kujundus on optimiseeritud erinevate seadmete jaoks. See tagab parema kasutajakogemus, kui kasutaja saab veebisaiti kasutada igas seadmes, ilma sisu kohandamata.

c) Kaardipõhine kujundus - on intuitiivne ja visuaalne võimalus informatsiooni ja andmete esitamiseks. Kaart kujutab endast ristkülikut, mille sisse on lisatud informatsioon. See võimaldab kasutajal kiirelt ja lihtsalt informatsioonile ligi pääseda.
d) Modaalsed aknad - on kasutajaliidese kujunduselement, mis võimaldab kuvada täiendavat sisu, ilma kasutajat uuele veebilehele suunamata. Seda võib kasutada lisainformatsiooni, aktsepteerimise teate kuvamiseks. Tegevuse jätkamiseks veebisaidil peab kasutaja kõigepealt avatud modaal akna sulgema.
e) Edusammude näitaja - on graafiline või numbriline kujunduselement, mis näitab mingi protsessi edusamme ehk kui kaugel ollakse eesmärgi täitmisest. Sellel abil saab kasutajat informeerida, kui palju on ülesandest täidetud, kui palju on ülesandeid alles või teisi üksikasju, mis on seotud ülesande ajaskaalaga.
f) Mikrointeraktsioonid - on oluline osa kasutajaliidese kujundusest, kuna aitavad luua kaasahaarava ja intuitiivse kasutajakogemuse. Enamasti on see seotud üksiku tegevusega nagu nupu vajutus või kursoriga elemendi kohalt üle liikumine, mis käivitab väikese animatsiooni või muutuse. Väikesed muutused võivad kasutajale pakkuda olulist tagasisidet mingi toimingu kinnitamisest või suunata kasutajat mingis protsessis.

Eelmainitud trendidest lähtuvalt on rakendus kujundatud selliselt, et kujundus oleks minimalistlik, lihtne kasutada, kujundus kohanduks kasutaja seadme jaoks ja sisaldaks endas mikrointeraktsioone.

3.1.2 Piltide eeltöötlus

Kasutajaliidese kaudu on võimalik anda töötlemiseks pilte, millel on kõigil erinevad dimensioonid ja resolutsioonid. Adobe [51] järgi kirjeldab pildi dimensioon pikslite arvu pildi kõrguse ja laiuse kohta. Resolutsioon kirjeldab, palju piksleid on ühe tolli kohta ning selle tähiseks on PPI (*pixels per inch*), mida rohkem piksleid ühe tolli kohta on seda kõrgem on resolutsioon ehk on rohkem detailsust. Allika järgi sõltub pildifaili suurus pikslite arvust pildil ning mida rohkem piksleid on pildil, seda mahukam on ka pildifail. Rakenduses on kasutajaliideses kasutatud pikslite arvu vähendamist, et pildifail muuta andmemahult väiksemaks ning seeläbi muuta pildi serverile töötlemiseks saatmine kiiremaks. Pildi pikslite arvu muutmiseks leitakse pildi laiuse pikslite arv, kui see ületab väärtust 1200 pikslit, siis muudetakse

pikslite arv 1200 peale ning kõrguse ja laiuse vahelise suhte säilitamiseks muudetakse ka pildi kõrguse pikslite arvu. Sama tehakse ka pildi kõrguse pikslitega, kui kõrguse piksleid on rohkem kui 800 pikslit, siis muudetakse kõrguse pikslite arv 800 peale ning pildi kõrguse ja laiuse suhte säilitamiseks muudetakse ka laiuse pikslite arvu.

Lisaks on olemas erinevad piltide kokkupakkimis formaadid nagu JPEG, PNG ja WebP, mis samuti mõjutavad pildifaili suurust. WebP [52] on moderne pildi formaat, mille abil saab pilte kokku pakkida ja kokkupakitud pilt on mahult väike. WebP kadudeta kokkupakitud pilt on 26% mahult väiksem, kui sama pilt PNG formaadis ja WebP kadudega kokkupakitud pilt on ligikaudu 30% mahult väiksem, kui JPEG formaadis pilt. Kuid WebP kokkupakkimist ei ole mõtet kasutada, kuna seda ei toeta veebibrauser Safari.

3.1.3 Kasutajaliidese tehnoloogia

Veebisait kasutab programmeerimiskeeli JavaScript, CSS (*Cascading Style Sheets*) ja HTML (*HyperText Markup Language*). Samuti on kasutatud kasutajaliidese raamistikku Vue, mis on JavaScript raamistik. Vue raamistik muudab kasutajaliidese reaktiivseks ehk automaatselt peab järge JavaScript staatuste muutuste kohta.

3.1.4 Kasutajaliidese funktsionaalsus

Kasutajaliidese funktsionaalsus kujutab endast funktsionaalsusi, mida kasutajal on võimalik läbi kasutajaliidese sooritada. Esimene funktsionaalsus on pildi valimine, mida kasutaja soovib anda rakendusele analüüsimiseks ja kirjelduse saamiseks. Kasutaja saab valida pildi oma seadmest lokaalselt salvestatud piltide hulgast nii mobiilivaates kui ka arvutivaates. Samuti on võimalik rakenduse abil teha pilti ning kasutada tehtud pilti kirjelduse saamiseks. Sobilikeks pildiformaatideks on PNG, JPEG ja WebP. Kui kasutaja on pildi valinud, toimub pildi eeltöötlus ning seejärel saadetakse pilt HTTP päringuga serverisse analüüsimiseks. Kui pildi analüüs serveri poolt on lõppenud, siis tagastatakse kasutajaliidesele pildi kirjeldus koos tuvastatud objektidega. Seejärel toimub pildi töötlemine, kus pildil tuvastatud objektide ümber tekitatakse kast koos objekti nimega, selleks on kasutatud veebitehnoloogiat Canvas API. Canvas API on mõeldud graafika joonistamiseks programmeerimiskeelte JavaScript ja HTML abil. Kui pildil on palju tuvastatud objekte ja kasutaja tahab tuvastatud objekti klasse üksikult vaadata, siis kasutajal

on võimalus värviliste piirikastide kuvamine sisse või välja lülitada rippmenüüs olevate valikute kaudu (vt joonis 12).

Joonis 12. Kasutajaliidese funktsionaalsus

3.2 Taustaprogramm ja funktsionaalsus

Allika [53] järgi on taustaprogrammi puhul tegemist rakenduse osaga, mis on vajalik sissetulevate päringute töötlemiseks ja päringule vastuse genereerimiseks ning kõik see toimub serveris, millega kasutaja suhtleb kasutajaliidese vahendusel. Taustaprogrammi jaoks leidub erinevaid raamistikke nagu Express JS, Diango, Flask ja Spring Boot. Raamistiku valiku puhul oluliseks teguriks on programmeerimiskeele Python tugi, sest vajalikud moodulid YOLOv8 ja EstNLTK tuginevad programmeerimiskeelel Python. Sellest tulenevalt sai valitud raamistik nimega Flask. Allika [54] järgi on Flask veebiraamistik, mis tähendab, et selles on olemas kõik vajalikud vahendid veebilehe ehitamiseks. Raamistik kuulub mikro-raamistike alla ehk tegu on raamistikuga, mis peaaegu üldse ei sõltu välistest teekidest. Allika järgi eelisteks on väheste väliste teekide pidev uuendamine ja teekidega seonduvate turvariskidega arvestamine.

3.2.1 Taustaprogrammi mudelid

Ennustuste tegemiseks taustaprogrammis on kasutatud kahte YOLOv8 mudelit. Ühe mudeli eesmärk on objektide tuvastamine ning teise mudeli eesmärk on rahakupüüride tuvastamine pildilt. Raamistikul YOLOv8 on erinevaid mudeleid objektide pildilt tuvastamiseks. Nendeks on YOLOv8n, YOLOv8s, YOLOv8m, YOLOv8l ja YOLOv8x, mis kõik on eeltreenitud COCO andmestikul [55]. Mudelite erinevuseks on treenitavate parameetrite ja sellest lähtuvalt ka erinev *mAP* väärtus (vt tabel 1). Mudeli andmete põhjal osutub kõige optimaalsemaks mudeliks YOLOv8m, kus parameetrite arv pole kõige kõrgem, kuid *mAP* väärtus on ligilähedane kõrgema parameetri arvuga mudelitele. Lisaks on mudeleid YOLOv8n, YOLOv8s ja YOLOv8m manuaalselt testitud samade testpiltidega, kus YOLOv8m oli kõige täpsemad objektituvastused. Sellest lähtuvalt sai valitud objektide tuvastamiseks YOLOv8m mudel.

Tabel 1. YOLOv8 mudelid koos parameetrite arvu ja *mAP* väärtusega [55].

Mudel	YOLOv8n	YOLOv8s	YOLOv8m	YOLOv8l	YOLOv8x
Parameetreid (miljon)	3.2M	11.2M	25.9M	43.7M	68.2M
mAP	37.3	44.9	50.2	52.9	53.9

Rahakupüüride tuvastamiseks on kasutatud andmestikku Euros Computer Vision Project [56], mis on saadud Roboflow keskkonnast. Andmestikus on treenimiseks ligikaudu 2600 pilti ning sellega on võimalik mudelit treenida tuvastama 5-, 10-, 20-, 50-, 100-, 200- ja 500-euroseid rahakupüüre. Ultralytics [57] järgi on soovitatud treenimiseks kasutada eeltreenitud objektituvastusmudelit, mis treenitakse omakorda uue andmestiku sisuga. Mudeli treenimiseks on kasutatud Google Colaboratory keskkonda, millel on võimalik treenimiseks kasutada graafilist kiirendit, et treenimine kulgeks kiiremini. Testimiseks sai treenitud YOLOv8m ja YOLOv8l mudelit. Testimise käigus selgus on YOLOv8l teeb mõningad juhul parema rahakupüüri tuvastuse kui YOLOv8m mudel. Sellest lähtuvalt sai valitud rahakupüüride tuvastamiseks YOLOv8l mudel, mis on treenitud rahakupüüride andmestikul.

3.2.2 Taustaprogrammi funktsionaalsus

Taustaprogramm saab kasutajaliideselt HTTP (*Hypertext Transfer Protocol*) päringuga pildi, millelt soovitakse tuvastada pildil olevad objektid ja pildi kirjeldus. Kõigepealt kontrollitakse, et saadetud pilt oleks sobilikus formaadis, sobivateks pildi formaatideks on PNG, JPEG ja WebP. Sellele järgneb pildi analüüsimine mooduli YOLOv8 abil. Pildil olevate objektide tuvastamiseks kasutatakse eeltreenitud mudelit YOLOv8m ning sellele järgneb rahakupüüride tuvastamine mudeliga YOLOv8l, mis on treenitud andmestikul nimega Euros Computer Vision Project [56]. Kui pildi analüüs on lõpuni jõudnud, siis tagastatakse tuvastatud objektid. Tuvastatud objektidel on koordinaadid, kus objekt pildil asub ning klassifikatsiooni indeks. Vastavalt klassifikatsiooni indeksile leitakse objekti eestikeelne nimi. Tuvastatud eestikeelsete nimedega koostatakse pildi kirjeldus, selleks loendatakse kokku, kui palju tuvastatud objekte pildil leidub ning leitud number muudetakse kirjakeelseks numbriks. Selleks, et objekti nimi sobiks kokku objekti pildil leidumise arvuga, tuleb muuta ka objekti käänet osastavasse käändesse. Kui objekti leidub üks eksemplar, siis on kasutatud ainsuse osastavat käänet ja kui mitu, siis on kasutatud mitmuse osastava käänet.

Sõna käänet on võimalik muuta raamistiku EstNLTK abil. EstNLTK [58] raamistik sisaldab endas naturaalse keele töötlemise funktsionaalsusi nagu paragrahvi, lause ja sõna tokeniseerimist, morfoloogilist analüüsi eesti keelele. Raamistiku meetodiga *vabamorf synthesize* abil saab muuta sõna vastavasse käändesse. Lõpuks koostatakse lause pildil olevate objektidest, kus igal objektil on arv, kui palju seda pildil esineb, koos õiges käändes objekti nimega. Kui kõik eelnev töötlemine ja andmed on kätte saadud, siis tagastatakse kasutajaliidesele pildi kirjeldus ja objektide paiknemise informatsioon.

3.3 Veebisaidi server

Allika [59] järgi on veebiserver tarkvara ja riistvara, mis kasutab HTTP protokolli päringuid ja teisi protokolle, et vastata kliendi päringutele, mis tulevad veebist. Peamiseks ülesandeks veebiserveril on kuvada veebisaidi sisu läbi veebilehtede hoiustamise, töötlemise ja kasutajatele pakkumisega. Veebiserveri riistvara on ühendatud internetiga ja tagab andmevahetuse teiste seadmetega. Allika järgi on veebiserver hea näide kliendi ja serveri vahelisest mudelist.

Rakenduses on kasutatud serveritarkvara nimega Ubuntu. Ubuntu on Linuxil põhinev operatsioonisüsteem. Ubuntule on installeeritud tarkvara nimega Apache, mille abil on võimalik pakkuda veebisisu interneti kaudu. Apache on veebiserver, mis töötleb ja pakub veebilehel olevat sisu HTTP protokolli kaudu [60]. Serveri IP (*Internet Protocol)* aadress on suunatud domeenile nimega estvision.ee ning serveri ja kliendi vaheline suhtlus on krüpteeritud SSL sertifikaadiga. Digicert [61] järgi on SSL (*Secure Socket Layer*) turvalisuse tehnoloogia, et luua krüpteeritud ühendus kliendi ja serveri vahel. Rakenduse taustaprogrammi funktsionaalsuse pakkumine toimub läbi veebiserverile määratud pordi, milleks on 3000. Cloudflare'i [62] järgi on port virtuaalne punkt, kus internetiühendused algavad ja lõppevad. Pordid on tarkvarapõhised ja hallatud arvutitarkvara poolt. Allika järgi on iga port seotud mingi kindla protsessi või teenusega. Taustaprogramm töötab Ubuntu tarkvaral protsessina.

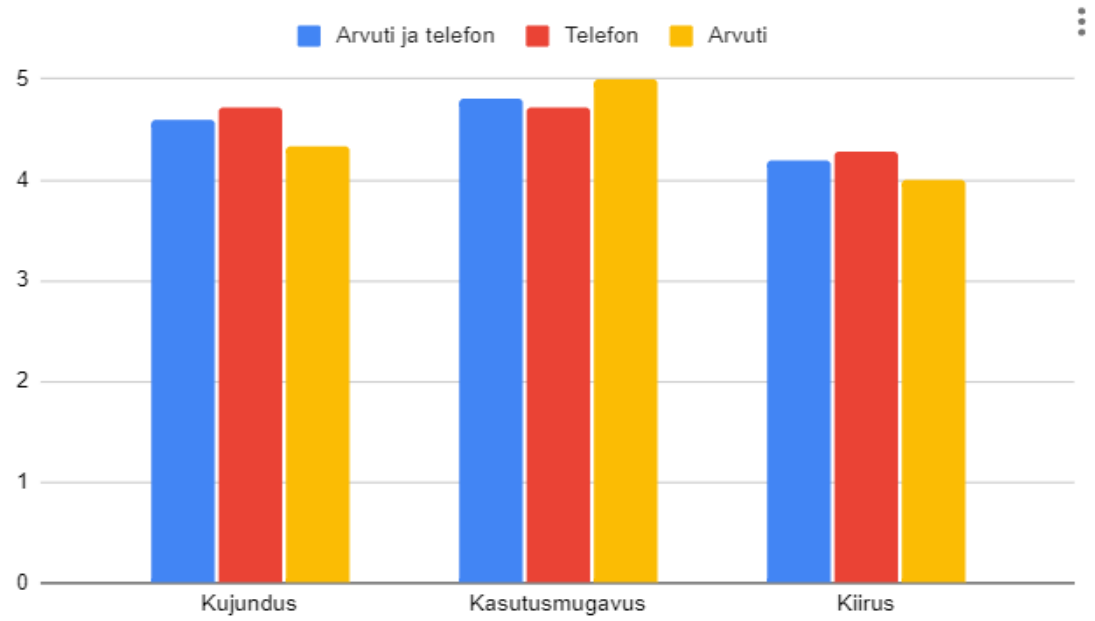
3.4 Rakenduse testimine

Artikli "What is...User testing" [63] järgi on kasutajatepoolne testimine protsess, mille käigus veebisaidi, rakenduse, toote või teenuse liides ja funktsioonid testitakse läbi päris kasutajate poolt, kes täidavad spetsiifilisi ülesandeid reaalsetes tingimustes. Allika järgi on testimise eesmärgiks hinnata veebisaidi või rakenduse kasutatavust ja hinnata, kas toode on valmis avalikuks kasutamiseks.

Testi eesmärk oli saada tagasisidet rakenduse kujunduse, kasutusmugavuse ja kiiruse osas, mis kulub rakendusel pildilt objektide tuvastamiseks, neid omadusi saab testija hinnata 5-palli skaalal (1 - kasin, 2 - rahuldav, 3 - hea, 4 - väga hea, 5 - suurepärane). Lisaks tuleb testijal testida, kui hästi suudab rakendus pildilt objekte tuvastada. Objektituvastuse testimiseks tuleb testijal teha rakenduse abil pilt objektist, mis leidub etteantud objektinimekirjas (vt lisa 1) ja sedasi vähemalt kümne erineva objekti suhtes. Seejärel tuleb ära märkida, kas pildil olevad objektid tuvastati täielikult, osaliselt või ei tuvastatud midagi. Testija saab ise valida, kas kasutab testimiseks nutitelefoni või arvutit.

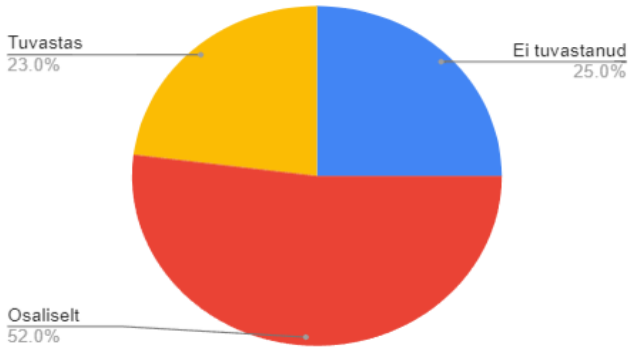
Rakendust testis 10 inimest vanusevahemikus 16 kuni 85. Lisaks anti rakendust kasutada 4-aastasele lapsele, testimise keskmiste tulemustes arvutamisel last ei arvestatud, kuna tema ei oska veel nii täpselt oma tagasisidet anda, küll aga nähtus lapse tagasisidest, et selline mudel võib lastele sobida.

Kõigi kasutajate poolt rakendusele antud hinnangute keskmised on välja toodud graafikus, kus kujunduse hinnang on 4.6, kasutusmugavuse hinnang on 4.8 ja kiiruse hinnang on 4.1 (vt joonis 13). Lisaks on eraldi välja toodud nii nutitelefoni kui ka arvutis kasutajate hinnangud rakendusele graafikus. Tagasiside põhjal sai kõige madalama keskmise hinnangu rakenduse kiirus, mis oleks võinud olla kõrgem.



Joonis 13. Hinnangute keskmised tulemused.

Objektituvastuse testimisel selgusid järgmised tulemused: kokku 100 pildi testimisel suutis rakendus objektid täielikult tuvastada 23-l pildil, osaliselt tuvastada 52-l pildil ja ei tuvastanud objekte 25-l pildil (vt joonis 14).



Joonis 14. Objektituvastamise testimise tulemused.

Osaliselt tuvastamisel rakendus kas ei tuvastanud pildil kõiki objekte või klassifitseeris objekti klassi valesti ja rakendus määras objektile mittevastava pealkirja (vt lisa 3). Lisas 2 on väljatoodud mõned näited, kus rakendus tuvastas, klassifitseeris ja määras pealkirja täpselt (vt lisa 2).

Kokkuvõte

Magistritöö eesmärgiks oli luua rakendus, mis kirjeldab pildi sisu eesti keeles. Rakenduse loomisega soovisin leida laste lugemaõppimist toetav lahendus. Laste motiveerimist pakkuva lahenduse mõte tuli ühest varasemast loost, kus käsitsi valmistatud pildi-sõna kaardikomplekt toetas lapse lugemaõppimist.

Eesmärgi saavutamiseks oli vaja uurida arvutinägemise tehnoloogiaid ja pildi kirjeldamise võimalusi. Arvutinägemise üheks osaks on objektituvastamine, mille abil on võimalik tuvastada pildil leiduvaid objekte. Objektituvastusmudelite arengut käsitledes otsustasin lähemalt vaadelda kaheastmelisi ja üheastmelisi mudeleid. Täpsemalt vaatlesin kahte kaheastmelist ja kolme üheastmelist objektituvastusmudelit. Kuna rakenduse puhul on ühe kriteeriumina oluline reaalajas vastuse saamine, siis seetõttu valisin vaadeldud mudelite hulgast üheastmelise mudeli. Valitud mudeliks on YOLO seeria kõige uuem mudel YOLOv8, mida on võimalik taustaprogrammile juurde lisada PIP paketina. Taustaprogrammi valiku kriteeriumiks oli programmeerimiskeele Python toetus, kuna YOLOv8 põhineb samuti sellel keelel. Sellest tulenevalt sai taustaprogrammi raamistikuks valitud raamistik Flask, mis seob kogu pildi analüüsimise üheks tervikuks ning tegeleb kasutaja poolt etteantud pildite analüüsi ja kirjelduse tagastamisega. Rakenduse puhul oli samuti oluline, et see oleks kasutatav nii nutitelefonis kui ka arvutis, selleks valisin kasutajaliideseks veebiliidese.

Rakendust anti testida kasutajatele, kes hindasid rakenduse kujundust, kasutusmugavust ja objektituvastuse kiirust 5-palli skaalal. Lisaks testisid kasutajd rakenduse objektituvastuse võimekust ise rakenduse kaudu pilte valides. Hindamisel vaadati, kas rakendus tuvastas pildil olevad objektid täielikult, osaliselt või ei tuvastanud üldse. Kasutajate keskmine hinnang kujundusele oli 4.6 punkti, kasutusmugavusele oli 4.8 punkti ja kiiruse hinnang oli 4.1 punkti. Kasutajad andsid positiivset tagasisidet rakenduse kujundusele ja kasutusmugavusele, kuid pildi kirjelduse koostamisel oleks võinud olla rakendus kiirem.

Objektituvastuse võimekuse hindamiseks testiti kokku 100 pilti, et teada saada kui hästi rakendus pildil olevaid objekte tuvastab. Rakendus tuvastas objektid täielikult 23-l pildil, 52-l pildil tuvastati objektid osaliselt ning 25-l pildil ei tuvastatud ühtegi objekti. Osalisel tuvastamisel kas klassifitseeriti objektide klassid valesti või ei tuvastatud kõiki objekte.

Perspektiivis soovin rakendust edasi arendada ja täiustada. Rakenduse mugavamaks kasutamiseks nutitelefonis võiks luua eraldi nutitelefoni rakenduse. Tuvastamisega seotud probleemid on võimalik lahendada mudeli treenimise kaudu. Väikelaste lugemist toetava funktsiooni arendamisel tuleks kuvatavate objektide pealkirju muuta väikelastele sobilikumaks rõõmsamate värvide kasutamise ja kujunduse kaudu. Esialgse põhifunktsiooni - tuvastatud objekti kohta info kuvamine eesti keeles - täiendavaks funktsiooniks võiks tulevikus olla ka info häälesitamine eesti keeles. Kuna analoogsed rakendused ei toeta eesti keelt, siis aitaks eesti keeles info kuvamine eristuda sarnastest rakendustest. Sellise lisafunktsiooniga rakenduse sihtgruppideks võiksid olla nägemispuudega inimesed. Nägemispuudega inimestele ei ole eesti keelt toetavaid laia funktsionaalsusega häälesitlusega rakendusi.

Kasutatud kirjandus

[1] Ren, J. & Wang, Y. 2022. Overview of Object Detection Algorithms Using Convolutional Neural Networks. *Journal of Computer and Communications*, 10, 115-132. https://www.scirp.org/journal/paperinformation.aspx?paperid=115011 (02.04.2023)

[2] Mishra, A. & Liwicki, M. 2019. Using Deep Features of Only Objects to Describe Images. Cornell University. https://arxiv.org/pdf/1902.09969.pdf (02.04.2023)

[3] Twitter. Help Center. How to write great image descriptions. https://help.twitter.com/en/using-twitter/write-image-descriptions (02.04.2023)

[4] Educasia. Describing picture and people. Thabyay Education Foundation. https://educasia.org/wp-content/uploads/RW-2-Describing-Pictures-and-People-Student.pdf (02.04.2023)

[5] Chen, A. UX Collective. 2020. How to write an image description. https://uxdesign.cc/how-to-write-an-image-description-2f30d3bf5546 (02.04.2023)

[6] Gaudenz Boesch. Viso.ai. The 100 Most Popular Computer Vision Applications in 2023. https://viso.ai/applications/computer-vision-applications/ (05.04.2023)

[7] Google Lens. What is Google Lens? https://lens.google/howlensworks/ (05.04.2023)

[8] Elise Williams. Wondershare. Use Google Lens OCR To Convert Images to Text https://pdf.wondershare.com/ocr/google-lens-ocr.html (07.04.2023)

[9] Microsoft. Seeing AI. https://www.microsoft.com/en-us/ai/seeing-ai (07.04.2023)

[10] Paths to Literacy. Seeing AI: Free App Narrating World Around You. https://www.pathstoliteracy.org/resource/seeing-ai-free-app-narrating-world-around-you/ (07.04.2023)

[11] Smartify. 2023. The ultimate cultural travel app. https://smartify.org/ (08.04.2023)

[12] Insider. 1 in 3 of babies are learning how to use smartphones before they can walk or talk. *The Journal.* 2015. https://www.businessinsider.com/one-third-of-babies-are-learning-how-to-use-smartphones-before-they-can-walk-or-talk-2015-4 (09.04.2023)

[13] Nicole Washington. PsychCentral. 31.07.2021. How Do Smartphones Affect Childhood Psychology?

https://psychcentral.com/lib/how-do-smartphones-affect-childhood-psychology#kids-and-phones (07.05.2023)
[14] Zou, Z., Chen, K., Shi, Z., Guo, Y., & Ye, J. 2019. Object Detection in 20 Years: A Survey. Cornell University. https://arxiv.org/pdf/1905.05055.pdf (12.04.2023)
[15] IBM. What is computer vision? https://www.ibm.com/topics/computer-vision (12.04.2023)
[16] Huang, T. S. 2016. Computer Vision: Evolution and Promise. 19th CERN School of Computing. http://cds.cern.ch/record/400313/files/p21.pdf 12.04.2023)
[17] Google Cloud. What is Artificial Intelligence (AI)? https://cloud.google.com/learn/what-is-artificial-intelligence (14.04.2023)
[18] Russell, S. J., Norvig, P. Artificial Intelligence. A Modern Approach, 3rd ed. Pearson Education, Inc. 2010. https://people.engr.tamu.edu/guni/csce421/files/AI_Russell_Norvig.pdf (14.04.2023)
[19] Sindhu, V., Nivedha, S., & Prakash, M. 2020. An empirical science research on bioinformatics in machine learning. ISSN (Online). https://jmcms.s3.amazonaws.com/wp-content/uploads/2020/02/24093934/6-AN-EMPIRICAL-SCIENCE-RESEARCH.pdf (15.04.2023)
[20] Mitchell, T. M. 1997. Machine Learning. McGraw-Hill Science/Engineering/Math. https://www.cin.ufpe.br/~cavmj/Machine%20-%20Learning%20-%20Tom%20Mitchell.pdf (15.04.2023)
[21] Alom, Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, S., Van Essen, B. C., Awwal, A. A. S., & Asari, V. K. 2018. The History Began from AlexNet: A Comprehensive Survey on Deep Learning Approaches. Cornell University. https://arxiv.org/ftp/arxiv/papers/1803/1803.01164.pdf (16.04.2023)
[22] IBM. What is a neural network? https://www.ibm.com/topics/neural-networks (16.04.2023)
[23] PyCode Mates. Multi-Layer Perceptron Explained: A Beginner's Guide. https://www.pycodemates.com/2023/01/multi-layer-perceptron-a-complete-overview.html?utm_content=cmp-true (18.04.2023)
[24] Rahul Awati. TechTarget. Convolutional neural network (CNN). https://www.techtarget.com/searchenterpriseai/definition/convolutional-neural-network (17.04.2023)

[25] Zhao, Z.-Q., Zheng, P., Xu, S., & Wu, X. 2016. Object Detection with Deep Learning: A Review. Cornell University. https://arxiv.org/pdf/1807.05511.pdf (17.04.2023)
[26] Javaid, S. AIMultiple. 2023. Quick Guide to Datasets for Machine Learning in 2023. https://research.aimultiple.com/datasets-for-ml/ (19.04.2023)
[27] Gauen, K., Dailey, R., Laiman, J., Zi, Y., & Asokan, N. Comparison of Visual Datasets for Machine Learning. *Loyola eCommons.* 2017. https://ecommons.luc.edu/cgi/viewcontent.cgi?article=1148&context=cs_facpubs (19.04.2023)
[28] Aziz, L., Salam, S. B. H., Sheikh, U. U., & Ayub, S. Exploring Deep Learning-Based Architecture, Strategies, Applications and Current Trends in Generic Object Detection: A Comprehensive Review. *Journals & Magazines.* 2020. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9186021 (21.04.2023)
[29] Everingham, M., S. M. Ali Eslami, S. M.A., Van Gool, L., Williams, C. K. I., Winn, J., Zisserman, A. 2014. The PASCAL Visual Object Classes Challenge: A Retrospective. Springer Science+Business Media. http://host.robots.ox.ac.uk:8080/pascal/VOC/pubs/everingham15.pdf (22.04.2023)
[30] Sharma, P. Analytics Vidhya. 2019. Computer Vision Tutorial: A Step-by-Step Introduction to Image Segmentation Techniques (Part) https://www.analyticsvidhya.com/blog/2019/04/introduction-image-segmentation-techniques-python/ (23.05.2023)
[31] ImageNet. ImageNet Large Scale Visual Recognition Challenge (ILSVRC). https://image-net.org/challenges/LSVRC/ (25.04.2023)
[32] Siyah, B. Kaggle. 2020. ImageNet Winning CNN Architectures (ILSVRC). https://www.kaggle.com/getting-started/149448 (25.04.2023)
[33] Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., Fei-Fei, L. 2015. ImageNet Large Scale Visual Recognition. Challenge. Cornell University. https://arxiv.org/pdf/1409.0575.pdf (24.04.2023)
[34] Tsang, S. H. *Data Science*. 05.04.2019. Review: GBD-Net/GBD-v1 & GBD-v2 - Winner of ILSVRC 2016 (Object Detection) https://towardsdatascience.com/review-gbd-net-gbd-v1-gbd-v2-winner-of-ilsvrc-2016-object-detection-d625fbeadeac (24.04.2023)
[35] COCO. COCO Common Objects in Context. https://cocodataset.org/#home (25.04.2023)

[36] Rahim, A., Maqbool, A., Rana, T. Plos One. 2021. Monitoring social distancing under various low light conditions with deep learning and a single motionless time of flight camera. https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0247440 (27.04.2023)
[37] Ciaglia, F., Zuppichini, F. S., Guerrie, P., McQuade, M., & Solawetz, J. 2022. Roboflow 100: A Rich, Multi-Domain Object Detection Benchmark. Cornell University. https://arxiv.org/pdf/2211.13523.pdf (08.04.2023)
[38] Koech, K. E. Towards Data Science. 2020. Object Detection Metrics With Worked Example. https://towardsdatascience.com/on-object-detection-metrics-with-worked-example-216f173ed31e (30.04.2023)
[39] Dubey, V. 2020. Evaluation Metrics for Object detection algorithms. https://medium.com/@vijayshankerdubey550/evaluation-metrics-for-object-detection-algorithms-b0d6489879f3 (30.04.2023)
[40] Girshick, R., Donahue, J., Darrell, T., & Malik, J. 2014. Rich feature hierarchies for accurate object detection and semantic segmentation. Cornell University. https://arxiv.org/pdf/1311.2524.pdf (08.04.2023)
[41] Zaidi, S. S. A., Ansari, M. S., Aslam, A., Kanwal, N., Asghar, M., & Lee, B. 2021. A Survey of Modern Deep Learning based Object Detection Models. Cornell University. https://arxiv.org/pdf/2104.11892.pdf (12.04.2023)
[42] Ren, S., He, K., Girshick, R., & Sun, S. 2016. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. Cornell University. https://arxiv.org/pdf/1506.01497.pdf (12.04.2023)
[43] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. 2015. You Only Look Once: Unified, Real-Time Object Detection. Cornell University. https://arxiv.org/pdf/1506.02640.pdf (12.04.2023)
[44] Awesome-yolo. Yolov8 (2023) developed by Ultralytics. https://github.com/srebroa/awesome-yolo (02.05.2023)
[45] Solawetz, J., Francesco, J. S. 2023. What is YOLOv8? The Ultimate Guide. roboflow. https://blog.roboflow.com/whats-new-in-yolov8/ (02.05.2023)
[46] The pip developers. Project description. https://pypi.org/project/pip/ (03.05.2023)

[47] Nikolaj Buhl. Encord. 2023. YOLO models for Object Detection Explained (Yolov8 Updated). https://encord.com/blog/yolo-object-detection-guide/#h7 (03.05.2023)
[48] Liu, W. 2016. SSD: Single Shot MultiBox Detector. Cornell University. https://arxiv.org/pdf/1512.02325.pdf (12.04.2023)
[49] Churchville, F. TechTarget. User Interface (UI). https://www.techtarget.com/searchapparchitecture/definition/user-interface-UI (27.04.2023)
[50] Bąk, T. Soft Kraft. 4 Timeless UI Design Examples to Inspire You in 2023. https://www.softkraft.co/ui-design-examples/#what-is-user-interface-design (27.04.2023)
[51] Adobe. 2022. Printed image resolution. https://helpx.adobe.com/photoshop/using/image-size-resolution.html (30.04.2023)
[52] WebP. An image format for the Web. https://developers.google.com/speed/webp (27.04.2023)
[53] Codecademy. Back-End Web Architecture. https://www.codecademy.com/article/back-end-architecture (27.04.2023)
[54] Pymbook. Introduction to Flask. https://pymbook.readthedocs.io/en/latest/flask.html#what-is-flask (27.04.2023)
[55] Ultralytics. Detect, Segment and Pose models are pretrained on the COCO dataset, while Classify models are pretrained on the ImageNet dataset. https://github.com/ultralytics/ultralytics (28.04.2023)
[56] Roboflow Universe. Euros Computer Vision Project. https://universe.roboflow.com/pp-deteccin-de-objetos/euros-7khiv (28.04.2023)
[57] Ultralytics. Ultralytics YOLOv8 Docs. https://docs.ultralytics.com/modes/train/ (28.04.2023)
[58] EstNLTK. Open source tools for Estonian natural language processing. https://estnltk.github.io (28.04.2023)
[59] Gillis, A. S. TechTarget. Web server. https://www.techtarget.com/whatis/definition/Web-server (30.04.2023)
[60] Sumo Logic. 2019. What is Apache? In-Depth Overview of Apache Web Server. https://www.sumologic.com/blog/apache-web-server-introduction/ (30.04.2023)
[61] DigiCert. What is an SSL Certificate? https://www.digicert.com/what-is-an-ssl-certificate (30.04.2023)

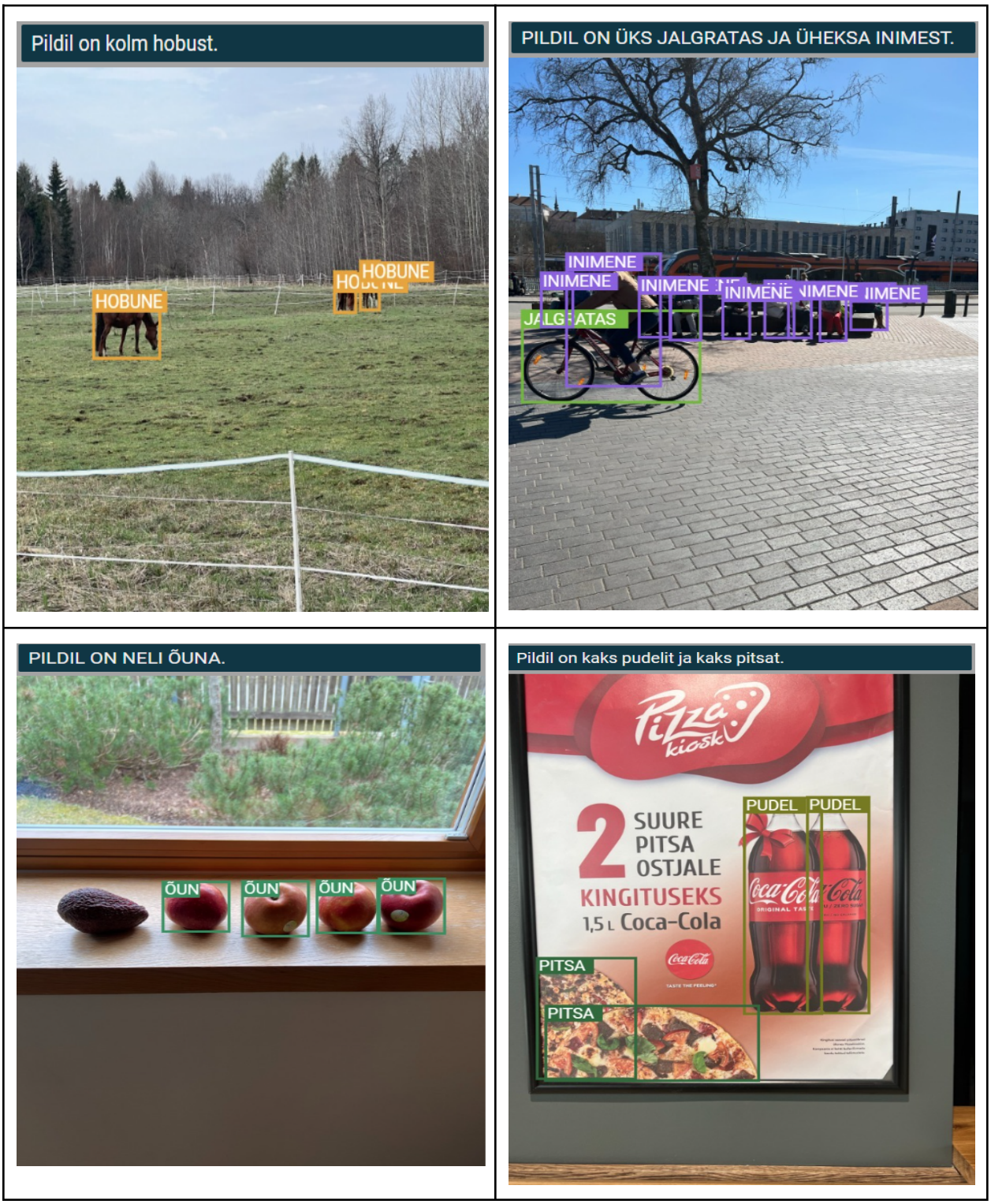
[62] Cloudflare. What is a computer port? Ports in networking. https://www.cloudflare.com/learning/network-layer/what-is-a-computer-port/ (30.04.2023)
[63] Omniconvert. What is...User testing. https://www.omniconvert.com/what-is/user-testing/ (05.05.2023)

Lisad

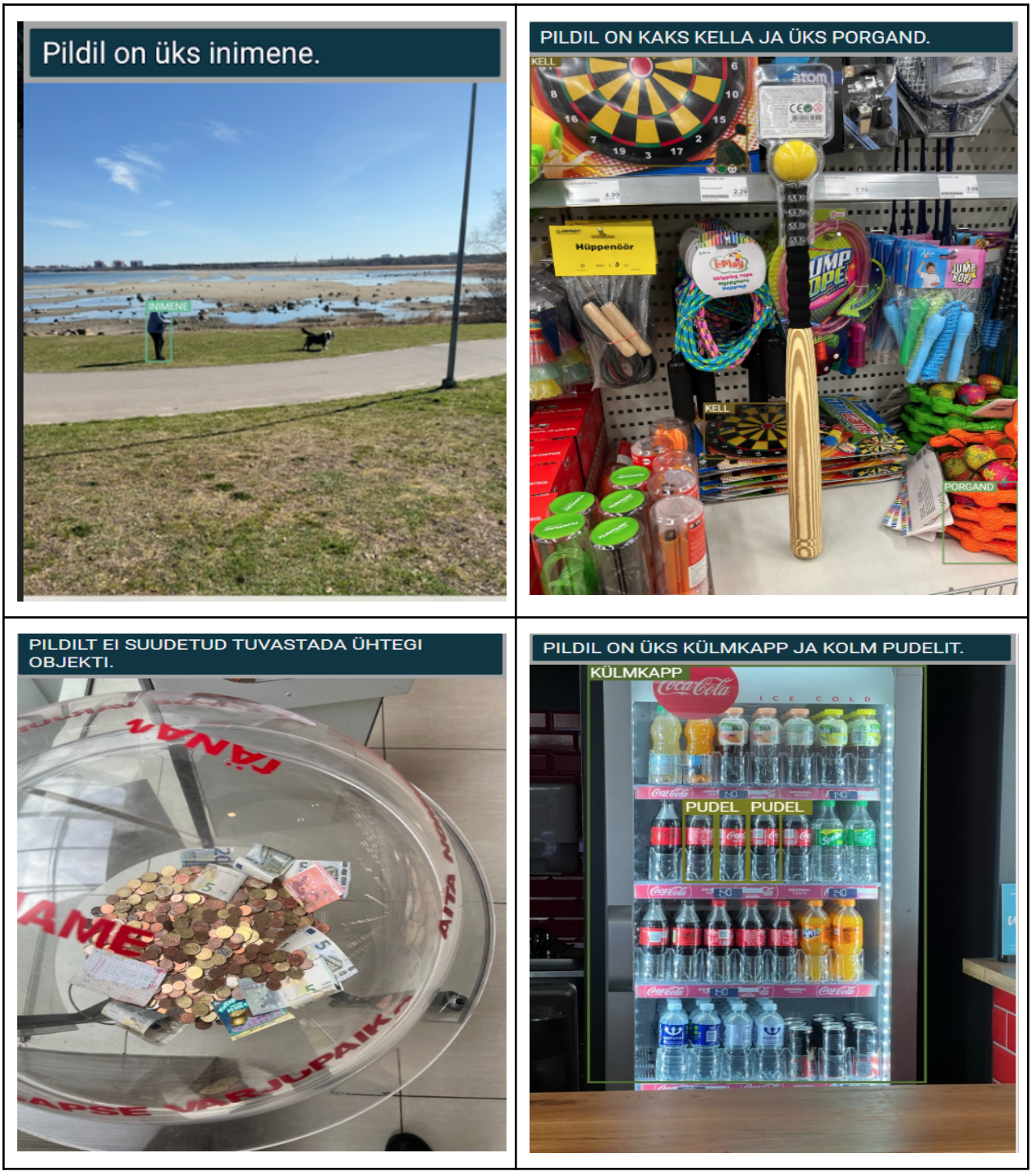
Lisa 1. Rakenduse tuvastatavate objektide nimekiri

a) Elusolendid: inimene, lind, kass, koer, hobune, lammas, lehm, elevant, karu, sebra, kaelkirjak;
b) Transpordivahendid: jalgratas, auto, mootorratas, lennuk, buss, rong, veoauto, paat;
c) Köögiviljad: banaan, õun, apelsin, brokkoli, porgand;
d) Toit: võileib, hotdog, pitsa, sõõrik, kook;
e) Söögivahendid: pudel, veiniklaas, tass, kahvel, nuga, lusikas, kauss;
f) Tänava elemendid: valgusfoor, tuletõrjehüdrant, parkimisautomaat;
g) Tehnikaseadmed: telekas, sülearvuti, arvutihiir, pult, klaviatuur, mobiiltelefon;
h) Spordivahendid: pesapallikurikas, pesapallikinnas, rula, surfilaud, tennisereket, lumelaud, spordiball, tuulelohe, suusad, frisbee;
i) Mööbliesemed: tool, diivan, voodi, söögilaud, pink;
j) Reisimine: seljakott, vihmavari, käekott, kohver;
k) Aksessuaarid: lips;
l) Taimed: toataim;
m) Vannitoa elemendid: wc-pott, kraanikauss;
n) Elektiseadmed: külmkapp, mikrolaineahi, ahi, röster, föön;
o) Esemed: raamat, kell, vaas, käärid, hambahari;
p) Mänguasjad: kaisukaru;
q) Märgid: stoppmärk;
r) Rahatähed: 5-eurone, 10-eurone, 20-eurone, 50-eurone, 100-eurone, 200-eurone, 500-eurone.

Lisa 2. Näited tuvastatud objektidest



Lisa 3. Näited objektide mittetuvastamisest ja valesti klassifitseerimisest




PILDIL ON ÜKS RAAMAT JA ÜKS 5EURONE.
5EURONE
RAAMAT


Pildil on üks auto ja üks külmkapp.
KÜLMKAPP
AUTO

Lihtlitsents lõputöö reprodutseerimiseks ja üldsusele kättesaadavaks tegemiseks

Mina, **Magnus Karlson**,

(autori nimi)

1. annan Tartu Ülikoolile tasuta loa (lihtlitsentsi) minu loodud teose

Piltide automaatne kirjeldamine eesti keeles,

(lõputöö pealkiri)

mille juhendaja on **Sven Aller**,

(juhendaja nimi)

reprodutseerimiseks eesmärgiga seda säilitada, sealhulgas lisada digitaalarhiivi DSpace kuni autoriõiguse kehtivuse lõppemiseni.

2. Annan Tartu Ülikoolile loa teha punktis 1 nimetatud teos üldsusele kättesaadavaks Tartu Ülikooli veebikeskkonna, sealhulgas digitaalarhiivi DSpace kaudu Creative Commonsi litsentsiga CC BY NC ND 4.0, mis lubab autorile viidates teost reprodutseerida, levitada ja üldsusele suunata ning keelab luua tuletatud teost ja kasutada teost ärieesmärgil, kuni autoriõiguse kehtivuse lõppemiseni.

3. Olen teadlik, et punktides 1 ja 2 nimetatud õigused jäävad alles ka autorile.

4. Kinnitan, et lihtlitsentsi andmisega ei riku ma teiste isikute intellektuaalomandi ega isikuandmete kaitse õigusaktidest tulenevaid õigusi.

Magnus Karlson

09.05.2023