

TALLINNA TEHNIKAÜLIKOOL  
Infotehnoloogia teaduskond

Caroly Huobolainen 155761IAPB

**MASINÕPPE ABIL HUUMORI  
TUVASTAMINE KARIKATUURIDE JA  
PORTREEDE NÄITEL**

Bakalaureusetöö

Juhendaja: Martin Rebane  
MSc

Tallinn 2018

## **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: Caroly Huobolainen

21.05.2018

## **Annotatsioon**

Lõputöö eesmärgiks on leida vastus küsimusele, kas masinõppe algoritmid on võimelised tuvastama huumorit piltidelt. Selle jaoks kasutatakse portreesid ja karikatuure inimestest.

Huumori tuvastamiseks kasutatakse kahte masinõppe algoritmi ja võrreldakse nende huumori tuvastamise suutlikust. Sealjuures viiakse läbi katseid sobliku treeningandmekogu leidmiseks, mille käigus analüüsitakse ka võimalikke probleeme andmekogu juures ja otsitakse neile lahendusi.

Eesmärgini jõudmisel kasutati baaskeelena Pythonit. Lisaks sellele on kasutusel ka Pythoni vabalt internetist kätte saadavad laiendused TensorFlow ja Keras.

Lõputöö on kirjutatud eesti keeles ning sisaldab teksti 31 leheküljel, 5 peatükki, 38 joonist.

## **Abstract**

### **Using machine learning to detect humor based on caricatures and portraits**

The goal of this thesis is to answer the question whether a machine learning algorithm is capable of detecting humor from pictures. In order to do so caricatures and portraits of persons are used. It is also discussed briefly on the topic of what makes a picture humorous – a detail that doesn't belong there or funny situation. Also the basics of machine learning, artificial intelligence and algorithms are discussed.

In the analysis part there are two algorithms used in order to detect humor and their performances are compared with one another. The analysis part of the thesis starts with finding the right dataset to use in comparing the two algorithms. In order to find correct dataset different sizes of datasets are evaluated. In addition to that the preprocessing of images is also thought of since in order to train machine learning algorithm the pictures have to be in same sizes.

In order to reach the goal of this thesis Python is used. In addition to Python, free libraries TensorFlow and Keras are being used.

The thesis is in Estonian and contains 31 pages of text, 5 chapters, 38 figures.

## Lühendite ja mõistete sõnastik

Ahenduskiht	<i>Pooling layer, subsampling layer</i> , konvolutsioonilise närvivõrgu kiht, mille eesmärgiks on pildi ruumiliste mõõtmete vähendamine järgmise kihi jaoks.
ANN	<i>Artificial neural network</i> , tehisnärvivõrk – masinõppe meetod, mis on disainitud jäljendama inimese aju informatsiooni töötlemist.
CNN	<i>Convolutional neural network</i> , konvolutsiooniline närvivõrk – masinõppemeetod, mida kasutatakse juhul kui sisendandmeteks on pildid. Disainitud jäljendama inimese aju informatsiooni töötlemist.
Kapselvõrk	<i>Capsule Network</i> , kapselvõrk – masinõppe meetod, mis kasutab neuronite asemel kapsleid. Kapsel suudab tuvastada pildilt objekti isegi siis kui see liigub pildil ringi või selle olek muutub, jättes objekti esinemise tõenäosuse samaks.
Konvolutsiooniline kiht	<i>Convolutional layer</i> , konvolutsioonilise närvivõrgu kiht, mille eesmärgiks on leida tunnusjooni pildilt kasutades filtreid.
Peidetud kiht	<i>Hidden layer</i> , tehisnärvivõrgu kiht, mis koosneb omavahel täielikult ühendatud neuronitest, millel on õpitavad kaalud ja vabaliikmed.
Ristentroopiakahju	<i>Cross-entropy loss</i> , ristentroopiakahjufunktsiooni rakendatakse konvolutsioonilise närvivõrgu väljundile peale seda kui see on kõik kihid läbinud. Sellega mõõdetakse klassifitseerimise mudeli jõudlust.
ReLU	<i>Rectified linear function</i> , mittenegatiivne lineaarfunktsioon – funktsioon, mida rakendatakse väljundile konvolutsioonilise kihi lõpus. Kui tulemus on väiksem kui 0, siis väljundiks on 0. Muul juhul on sisend funktsiooni väljundiks.
SNN	<i>Spiking neural network</i> , rikastatud närvivõrk – masinõppe meetod, mis on inspireeritud bioloogilistest närvivõrkudest. Toimivad kasutades rikastatud neuroneid, mis on diskreetsed sündmused ning toimuvad kindlatel ajahetkedel, mitte koguaeg.
Tunnuste kaart	<i>Feature map</i> , konvolutsioonilise närvivõrgu ühe kihi väljund, mis moodustatakse libistades filtrit üle pildi ja arvutades punkti tulemuse.
Täielikult ühendatud kiht	<i>Fully connected layer</i> , konvolutsioonilise närvivõrgu kiht, mille eesmärgiks on klassifitseerida pilte.

## Sisukord

1 Sissejuhatus .....	10
2 Huumor.....	12
2.1 Portree ja karikatuur .....	12
3 Masinõpe .....	14
3.1 Tehisnärvivõrgud.....	14
3.1.1 Neuron .....	15
3.2 Konvolutsioonilised närvivõrgud .....	16
3.2.1 Konvolutsiooniliste närvivõrkude arhitektuur.....	16
3.2.2 Konvolutsiooniliste närvivõrkude treenimine .....	18
3.3 Rikastatud närvivõrgud.....	19
3.4 Kapselvõrgud.....	19
3.4.1 Kapsel .....	20
3.4.2 Kapselvõrgu arhitektuur .....	21
3.5 Kasutatud tehnoloogiad .....	23
3.5.1 Keras.....	23
3.5.2 TensorFlow.....	23
4 Eksperimendid .....	24
4.1 Andmekogum .....	24
4.2 Piltide eeltötlus .....	25
4.3 Konvolutsiooniline närvivõrk.....	27
4.3.1 Konvolutsioonilise närvivõrgu treenimine .....	28
4.3.2 Mudel 50 portree ja 50 karikatuuriga, piltide laiuse ja kõrguse suhted muudetud .....	28
4.3.3 Mudel 50 portree ja 50 karikatuuriga, piltide laiuse ja kõrguse suhted muutmata .....	31
4.3.4 Mudel 200 portree ja 200 karikatuuriga, piltide laiuse ja kõrguse suhted muutmata .....	33
4.3.5 Järeldused .....	35
4.4 Kapselvõrk.....	36

4.4.1 Kapselvõrgu treenimine.....	37
4.4.2 Mudel 50 portree ja 50 karikatuuriga .....	37
4.4.3 Mudel 200 portree ja 200 karikatuuriga .....	38
4.4.4 Järeldused .....	38
5 Kokkuvõte .....	41
Lisa 1 – Treeningandmekogum .....	45

## Jooniste loetelu

Joonis 1. Portree Donald J. Trumpist [11].....	13
Joonis 2. Karikatuur Donald J. Trumpist [12].....	13
Joonis 3. Tehisnärvivõrgu arhitektuur (autori joonis). ....	14
Joonis 4. Neuron (autori joonis). ....	15
Joonis 5. Konvolutsioonilise närvivõrgu arhitektuur [21].....	16
Joonis 6. Värviline (sisend)pilt (autori joonis). ....	17
Joonis 7. Mittenegatiivne lineaarfunktsioon (autori joonis). ....	18
Joonis 8. Softmax funktsioon (autori joonis).....	18
Joonis 9. Kapsel (autori joonis). ....	20
Joonis 10. Kapselvõrgu arhitektuur (autori joonis). ....	21
Joonis 11. Sigmoid funktsioon (autori joonis). ....	22
Joonis 12. Portree Barack H. Obamast [39]. ....	24
Joonis 13. Karikatuur Barack H. Obamast [12].....	24
Joonis 14. Katikatuur Adam Schiffist [12].....	25
Joonis 15. Karikatuur Joe Bidenist [12]. ....	25
Joonis 16. Ruudukujuline sisendpilt Adam Schiffist [12], autori markeeringuga.....	26
Joonis 17. Ruudukujuline sisendpilt Joe Bidenist [12], autori markeeringuga. ....	26
Joonis 18. Ruudukujuline pilt mustade äärtega Adam Schiffist [12], autori markeeringuga. ....	27
Joonis 19. Ruudukujuline pilt mustade äärtega Joe Bidenist [12], autori markeeringuga. ....	27
Joonis 20. Töös kasutatava konvolutsioonilise närvivõrgu arhitektuur (autori joonis)..	28
Joonis 21. Mudeli täpsuse graafik (autori joonis). ....	29
Joonis 22. Mudeli kao graafik (autori joonis).....	29
Joonis 23. Kõrgeima tõenäosusprotsendiga portree [40], autori markeeringuga. ....	30
Joonis 24. Kõrgeima tõenäosusprotsendiga valesti klassifitseeritud portree [41], autori markeeringuga. ....	30
Joonis 25. Kõrgeima tõenäosusprotsendiga karikatuur [12], autori markeeringuga. ....	31



Joonis 26. Kõrgeima tõenäosusprotsendiga valesti klassifitseeritud karikatuur [12], autori markeeringuga.....	31
Joonis 27. Kõrgeima tõenäosusprotsendiga portree [42], autori markeeringuga. ....	32
Joonis 28. Kõrgeima tõenäosusprotsendiga vale kategooriaga portree [43], autori markeeringuga. ....	32
Joonis 29. Kõrgeima tõenäosusprotsendiga karikatuur [13], autori markeeringuga. ....	33
Joonis 30. Kõrgeima tõenäosusprotsendiga vale klassiga karikatuur [13], autori markeeringuga. ....	33
Joonis 31. Mudeli täpsuse graafik (autori joonis). ....	34
Joonis 32. Mudeli kao graafik (autori joonis).....	34
Joonis 33. Konvolutsioonilise närvivõrgu mudelite tulemuste võrdlus (autori joonis)..	35
Joonis 34. Karikatuur, mille klassifitseerisid valesti kõik kolm mudelit [13]. ....	36
Joonis 35. Töös kasutatava kapselvõrgu arhitektuur (autori joonis). ....	37
Joonis 36. Õigesti klassifitseeritud karikatuur [13]. ....	38
Joonis 37. Valesti klassifitseeritud piltide võrdlus mudelite kaupa (autori joonis).....	39
Joonis 38. Karikatuur, mida ei tundnud ära kumbki kapselvõrk [12]. ....	40

## 1 Sissejuhatus

Huumor ja sellega kaasnev naer on olulised naudingu allikad ning inimesed, kes suudavad pakkuda seeläbi teistele naudingut, on nende kaaslaste jaoks meelepärased [1]. Huumoriga kaasneb heaolutunne, sest selle peamiseks funktsiooniks on vabastada emotsioone, arvamusi või tundeid, mis omakorda mõjuvad positiivselt inimese tervisele [2]. Täiskasvanud inimene naerab keskmiselt 18 korda päevas [3], mis tähendab, et huumoril on tähtis osa inimeste igapäevaeludes. Kuna huumor sõltub inimeste kultuurilisest taustast, sotsiaalsesse gruppi kuulumisest ning perekonnast ja selle vaadetest, siis inimesed kellel on sarnane huumorisoon, leiavad kiiremini ja kergemini ühise keele. Järelkult on huumor subjektiivne ning raske on hinnata, mis konkreetset teeb teksti või pildi naljakaks. Küll aga peetakse humoorikaks enamasti asju, mis ei kattu sellega, mida arvatakse juhtuvat [4]. Huumor on nii suusõnaline, kirjalik kui ka visuaalne. Antud töös tegeletakse visuaalse huumoriga.

Visuaalse huumori mõistmine on täis väljakutseid, sest ühes pildis leidub väga palju informatsiooni. Inimese puhul piisab kiire pilgu heitmisest pildile, et tuua välja tohutu hulk üksikasju stseeni kohta [5]. Visuaalse huumori tuvastamiseks tuleb pildilt leida kõik objektid ja jälgida nende koos toimimist ning mõista pildi konteksti [3]. Tuleb süveneda pildi detailidesse, mõelda selle dünaamikale, teha järeldusi, kuidas olukord areneks järgmise paari sekundi jooksul, kuidas situatsioon areneks pildil olevate inimeste mõtetes ning sealjuures põhjendada oma peas ka kui tõenäoline oleks taoline käitumine pildil olevate inimeste poolt [6]. Huumoril on palju eri liike nagu näiteks sõnamäng, ironia, sarkasm, kuid huumori kohta on olemas vähe süstemaatilisi omadusi. Huumor on alati asetatud suuremasse konteksti, mis võib aeg-ajalt tähendada, et huumori täielikuks mõistmiseks, on vaja teada palju selle tausta kohta [7]. Seetõttu on raske hinnata, mis muudab ühe pildi naljakaks ning sealjuures seda õpetada ka tehisintellektile. Kui arvuti suudaks ära arvata inimkeeles leiduva huumori, suudaks see paremini mõista inimkeele tõelist tähendust ja seeläbi teha paremaid otsuseid, mis parandaks omakorda kasutajakogemust [7]. Arvutuslik visuaalne huumor on kasulik

mitmete rakenduste jaoks: paremate fotode redigeerimiste vahendite loomiseks, nutikate kaamerate jaoks, mis valivad õige (naljaka) hetke pildi tegemiseks, vahendid naljakate piltide kõrgemalt hindamiseks (näiteks sotsiaalmeediasse postitamisel), videote kokku võtmise vahend toomaks välja naljakad kaadrid, automaatselt naljakate stseenide genereerimine meelelahutuseks, isikupärastatud huumori identifitseerimiseks ja pakkumiseks jne [3].

Antud töö eesmärgiks on välja selgitada masinõppemeetodite efektiivsus huumori tuvastamisel kasutades portreesid ja karikatuure. Analüüsi osas keskendutakse kahele masinõppemeetodile - konvolutsioonilistele närvivõrkudele ja kapselvõrkudele. Masinõppemeetodite valiku tegemisel lähtuti sellest, et konvolutsioonilised närvivõrgud on väga populaarne pildituvastuse masinõppemeetod ning kapselvõrgud peaksid olema antud ülesande lahendamisel efektiivsemad.

## 2 Huumor

Huumor on oluline osa inimeste vahelises suhtluses. Sarnase huumorimeelega inimesed leiavad kergelt ühise keele ning jututeemad. Huumoril on mitmeid erinevaid vorme – see võib olla nii suuline, kirjalik kui ka visuaalne. Peamisteks huumoriteooriateks on üliluslikkuse teooria (*the Superiority Theory*), mis ütleb, et naeru põhjus on paremuse tundmine, leevendusteooria (*the Relief Theory*), mis ütleb, et naer on närvilise energia vabanemine ning ebakõla teooria (*the Incongruity Theory*), mis ütleb, et see on midagi kohatut ehk midagi, mis rikub meie vaimseid mustreid ja ootusi. Ebakõla teooria lähenemise võtsid omaks nii James Beattie, Immanuel Kant, Arthur Schopenhauer, Søren Kierkegaard kui ka paljud hilisemad filosoofid ja psühholoogid. Praegu on ebakõla teooria filosoofias ja psühholoogias domineerivaks huumoriteooriaks [8]. Universaalset huumori definitsiooni on aga raske saavutada, sest erinevad inimesed saavad isegi samast lausest erinevalt aru [7]. Kõige lihtsam viis huumori defineerimiseks on, et huumoriks nimetatakse kõike, mida inimesed peavad naljakaks. See, mida peetakse naljakaks on väga palju mõjutatud inimese biofüüsikalisest, psühholoogilisest, sotsiaalkultuurilisest ja vaimsest seisundist. Intsident, mis ühe inimese arvates on lõbus, võib olla teise inimese arvates solvav, maitsetu või emotsionaalselt valus [9]. Järelikult on väga raske hinnata, mis on naljakas ja mis ei ole.

### 2.1 Portree ja karikatuur

Portree on pilt, foto või skulptuur millel kujutatakse inimest väga sarnaselt reaalsusele, mille peamiseks ideeks on tuua välja sellel kujutatava persooni olemus. Enamasti jäädvustatakse sellele inimese nägu ning ülakeha. Karikatuur on humoorikas ja moonutatud pilt, mis on üks visuaalse huumori liikidest. Karikatuur nagu ka iga portree, on katse näidata pildil kujutatava inimese olemust [10]. Karikatuuri joonistamisel on autorile antud vabamad käed, autor ei pea tegema kujutatavast persoonist identset pilti, vaid tihtipeale üritatakse eriti täpselt välja tuua mingisuguseid inimest iseloomustavaid jooni. Karikatuurid näitavad vaatajale sel kujutatava inimese identiteeti kasutades sübolismi või piltlikku liialdust [10]. Karikatuuril võib olla kujutatud inimese nägu

suuremana tegelikkusest, keha ning käed palju väiksemana ning lisatud on ilme, mis vastav ta tavapärasele hoiakule.

President Donald J. Trumpi portree (Joonis 1) on autor jäädvustanud tema parimad küljed, isiksuse ja väga õnneliku ning heatahtliku ilme.



Joonis 1. Portree Donald J. Trumpist [11].

Vastukaaluks on aga karikatuuril (Joonis 2) kujutatud Donald J. Trumpi väga üleoleva, enesekeskse ning õela ilmega, sealjuures on joonistatud ta pea ebaproportsionaalselt suureks ning keha kujutatud palju väiksemana



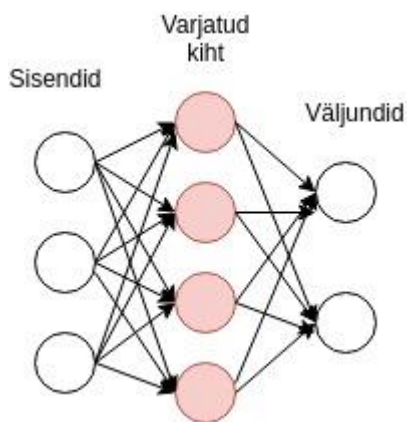
Joonis 2. Karikatuur Donald J. Trumpist [12].

### 3 Masinõpe

Tehisintellekt on inimese mõtteprotsesside simulatsioon masinate, eelkõige arvutisüsteemide poolt. Nende protsesside hulka kuuluvad õppimine (teabe hankimine ja selle kasutamine), põhjendamine (reeglite kasutamine jõudmaks ligikaudsete või kindlate järeldusteni) ja iseenda parandamine [13]. Masinõpe on tehisintellekti rakendus, mis annab süsteemidele võime automaatselt õppida ja täiustuda läbi kogemuste ilma inimese sekkumiseta [14]. Masinõpet kasutatakse internetis laialdaselt nii otsingute vastete leidmisel, kõne tekstiks töötlemisel kui ka näiteks näo tuvastamiseks piltidelt. Sügavõpe on masinõppe klass, mis tegeleb hierarhiliste abstraktsioonide modelleerimisega sisendandmetes mitmete kihtide abil [15]. See on arengujärgus lähenemisviis ning seda kasutatakse laialdaselt ka traditsioonilises tehisintellekti valdkondades nagu semantiline analüüsimine, siirdeõpe, naturaalse keele töötlemine, tehisnägemine jne [16]. Antud lõputöös on masinõppemeetoditest kasutusel konvolutsioonilised närvivõrgud ja kapselvõrgud.

#### 3.1 Tehisnärvivõrgud

Tehisnärvivõrgud (*Artificial Neural Network, ANN*) on üks levinud meetoditest masinõppes. Tehisnärvivõrgud bioloogiast inspireeritud arvutiprogrammid disainitud jäljendama inimese aju informatsiooni töötlemist [17]. Närvivõrgud koosnevad sisendandmetest, ühest või rohkemast varjatud kihist (*hidden layer*) ja ühest väljundkihist (Joonis 3).



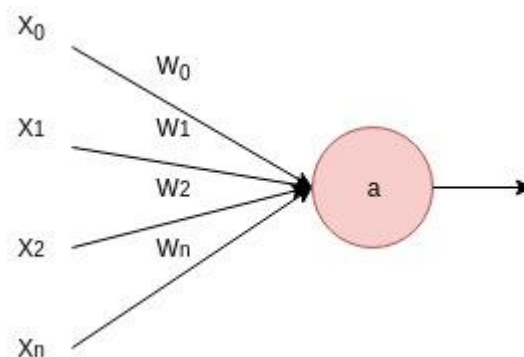
Joonis 3. Tehisnärvivõrgu arhitektuur (autori joonis).

Varjatud kihid on omavahel enamasti täielikult ühendatud ehk kõik neuronid on järgmise kihi neuronitega ühenduses. Neuronid, mis on joonisel kujutatud punastena, on omavahel täielikult ühendatud ja neil kõigil on õpitavad kaalud ning vabaliikmed [18]. Tehisnärvivõrgu sisendid läbivad varjatud kihid ning jõuavad väljundkihini, kus tehakse järeldused. Väljundkihi neuronite arv sõltub probleemist.

Tehisnärvivõrgud koguvad oma teadmised leides andmetest mustreid ja suhteid ja nad õpivad (või neid treenitakse) läbi kogemuste, mitte läbi programmeerimise [17]. Tehisnärvivõrkude puhul ei ole efektiivne pilte sisendina kasutada, sest sel juhul suureneks kaalude arv järsult, kuna igat piksel vajaks oma neuronit sisendkihil [18].

### 3.1.1 Neuron

Neuron on tehisnärvivõrgu kõige väiksem osake. Neuron saab sisendi eelmise kihi neuronitelt, seejärel korrutab selle skalaarkaalude ja summadega. See summa antakse edasi aktiveerimisfunktsioonile, mis väljastab skalaari, mis omakorda on järgmise kihi neuroni sisendiks [19]. Aktiveerimisfunktsiooniks võib olla Gaussi funktsioon, logistiline funktsioon, hüperboolne funktsioon või näiteks lineaarne funktsioon [20].



Joonis 4. Neuron (autori joonis).

Joonisel 4 on kujutatud tüüpilist neuronit, kus  $x_1, x_2, \dots, x_n$  on neuroni sisendid,  $x_0$  on vabaliige,  $w_1, w_2, \dots, w_n$  on kaalud ning  $a$  on neuroni väljundfunktsioon.

Sisendid on kas sisendkihi väärtused või mõne peidetud kihi väärtused. Vabaliige on konstant, mis lisatakse aktiveerimisfunktsiooni sisendisse.

Valem (1) esitab neuroni väljundi  $f$  aktiveerimisfunktsiooni, mis muudab närvivõrgu väga paindlikuks ja annab talle võime hinnata keerulisi mittelineaarseid suhteid andmetes [20].

$$a = f(\sum_{i=0}^N w_i x_i) \quad (1)$$

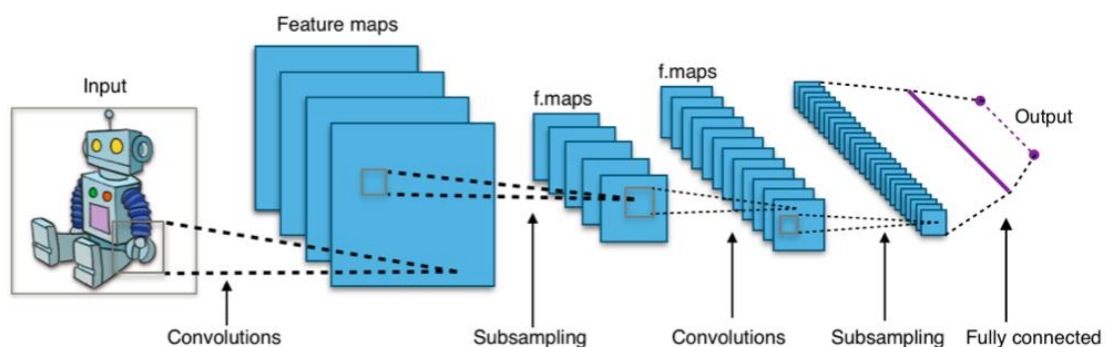
## 3.2 Konvolutsioonilised närvivõrgud

Konvolutsioonilistel närvivõrkude (*Convolutional Neural Network, CNN*) arhitektuur sarnaneb tehisnärvivõrkudele, millele on lisandunud kaks kihti. Nendeks on konvolutsiooniline kiht (*convolutional layer*) ja ahenduskiht (*pooling layer, subsampling layer*), mis asetsevad närvivõrgus kohe peale sisendkihti. Konvolutsioonilisi närvivõrke kasutatakse enamasti juhul kui sisendandmeteks on pildid.

Konvolutsiooniliste närvivõrkude ülesanne on tuvastada pildi pikslite olulisi funktsioone. Nende jaoks ei ole oluline, kus asub kindel objekt, vaid kas see asub pildil. Seetõttu suudavad konvolutsioonilised närvivõrgud pildilt tuvastada näo, silmad, suu, nina, kõrvad ja teised näo komponendid, kuid nad ei arvesta eri komponentide omavaheliste suhetega. Ehk kui näol oleks silmad, nina ja suu vales järjekorras, peaks konvolutsiooniline närvivõrk seda siiski näoks. Järelikult sobivad konvolutsioonilised närvivõrgud portreede ja karikatuuride tuvastuseks, kuid tulemused ei pruugi olla väga täpsed.

### 3.2.1 Konvolutsiooniliste närvivõrkude arhitektuur

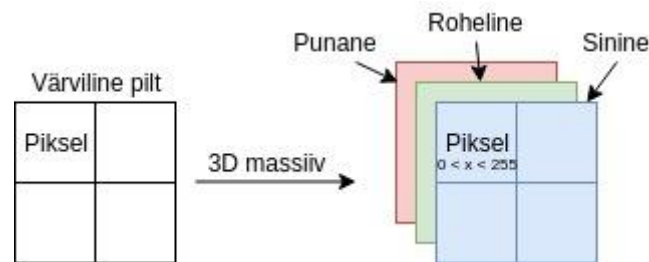
Konvolutsiooniline närvivõrk koosneb kolmest kihist: konvolutsioonilisest kihist, ahenduskihist ja täielikult ühendatud kihist (*fully connected layer*). Konvolutsioonilisi ja ahenduskihte võib närvivõrgus olla rohkem kui üks (Joonis 5).



Joonis 5. Konvolutsioonilise närvivõrgu arhitektuur [21].



Konvolutsioonilise närvivõrgu sisendina kasutatakse enamasti pilte. Iga pilt koosneb pikslitest. Must-valgete piltide puhul on konvolutsioonilise närvivõrgu sisendiks kahedimensiooniline massiiv, kus iga piksel näitab värvi intensiivsust ning on vahemikus 0-255. Null tähistab valget värvi ning kakssada viiskümmend viis musta. Selle informatsiooni põhjal on konvolutsiooniline närvivõrk võimeline tegema järeldusi. Värviliste piltide puhul on sisendiks kolmedimensiooniline massiiv, kus iga piksli kohta on kolm väärtust, mis näitavad kas punase, roheline või sinise värvi intensiivsust (Joonis 6).

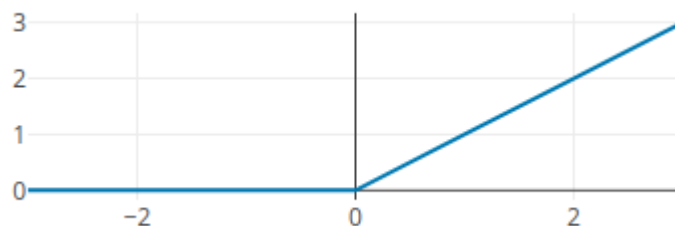


Joonis 6. Värviline (sisend)pilt (autori joonis).

Nii nagu ka must-valgete piltide puhul, on piksli võimalikud väärtused, mis on joonisel tähistatud  $x$ 'iga, vahemikus 0-255. Konvolutsioonilise närvivõrgu sisendkiht koosneb neuronitest, mis kodeerivad sisendpikslite väärtusi [22]. Sellele järgneb konvolutsiooniline kiht, mis koosneb õpitud filtrite kogumist (Joonis 6) [23]. Filtrid on väga väikesed matriksid, mis käivad läbi terve sisendpildi ning aktiveeruvad ainult juhul kui nad leiavad otsitava tunnusoone pildilt. Konvolutsioonilise närvivõrgu eesmärgiks on leida kõige efektiivsemad filtrid antud andmestiku jaoks. Närvivõrk õpib nägema teatud tüüpi visuaalseid funktsioone, näiteks mõne suuna või värvi esimest kihti ning tulemuseks annavad nad tunnuste kaardi (*feature map*) [23]. Tunnuste kaardid tähistavad aktiveeritud piirkondi ehk siis regioone, kus on tuvastatud filtrile iseloomulikud funktsioonid. Filtri matriksi väärtused muutuvad iga õppimisiteratsiooni käigus, mis näitabki, et võrk õpib tuvastama, millised piirkonnad andmetest on funktsioonide jaoks olulised [24]. Konvolutsioonilise kihi lõpus rakendatakse väljundile mittenegatiivset lineaarfunktsiooni (*ReLU*), mis on esitatud valemiga (2). See tähendab, et kui tulemus on väiksem kui 0, siis väljundiks on 0, muul juhul funktsiooni väljundiks sisendväärtus [25].

$$f(x) = \max(x, 0) \tag{2}$$

Graafikul näeks mittenegatiivne lineaarfunktsioon järgmine (Joonis 7):



Joonis 7. Mittenegatiivne lineaarfunktsioon (autori joonis).

Ahenduskiht asub tavaliselt peale konvolutsioonilist kihti ning selle esmaseks eesmärgiks on ruumiliste mõõtmete (laiuse ja kõrguse) vähendamine järgmise konvolutsioonilise kihi jaoks [24]. Täelikult ühendatud kihis rakendatakse *softmax* aktiveerimisfunktsiooni, mis viib iga väljundväärtuse vahemikku 0 kuni 1 selliselt, et kõikide väljundite summa on 1. Valem (3) ehk *softmax* aktiveerimisfunktsioon on  $z$  väljundkihi sisendite vektoriks ning  $j$  indekseerib väljundeid ehk siis  $j = 1, 2, \dots, K$ .

$$\sigma(z)_j = \frac{e^{z_j}}{\sum_{k=1}^K e^{z_k}} \quad (3)$$

Graafikul näeks *softmax* funktsioon välja järgmine (Joonis 8):



Joonis 8. *Softmax* funktsioon (autori joonis).

### 3.2.2 Konvolutsiooniliste närvivõrkude treenimine

Konvolutsioonilise närvivõrgu treenimisel läbib pilt kõik kihid ning seejärel teeb võrk ennustuse, mis pildiga oli tegu. Peale seda kasutatakse ristentroopiakahju (*cross-entropy loss*) funktsiooni, et mõista kui hästi võrk toimib. Ristentroopiakahju või teise nimega logaritmiline kahju, mõõdab klassifitseerimise mudeli jõudlust, mille väljund on tõenäosusväärtus vahemikus 0 ja 1. Ristentroopiakahju suureneb juhul kui ennustatud tulem erines tegelikust [26]. Valem (4) esitab ristentroopiakadu, mille leidmiseks on vaja teada ennustatud väärtust  $q$  ja tegelikku tulemust  $p$ .

$$H(p, q) = -\sum_x p(x) \log(q(x)) \quad (4)$$

Kui viga on arvatud, siis pöörduv tulemus tagasi üle võrgu ning peale seda reguleeritakse vastavalt kaalusid. Ehk siis igal õppimisiteratsioonil muudetakse neuronite vahelist kaalu vähesel määral selles suunas, mis viga vähendaks, ja kui jõuame punkti, kus mõlemale poole liikudes viga suureneb oleme jõudnud optimumi. Tavaliselt proovitakse õppimisprotsessi mitu korda ja valitakse parim optimum [27].

### 3.3 Rikastatud närvivõrgud

Rikastatud närvivõrgud (*Spiking Neural Network, SNN*) nagu ka teised närvivõrgud on inspireeritud bioloogilistest närvivõrkudest ning ka nende kõige väiksemateks osakesteks on neuronid. Rikastatud närvivõrgud erinevad tavapärastest närvivõrkudest sellepolest, et opereerivad kasutades rikastatud neuroneid, mis omakorda on diskreetsed sündmused ning toimuvad kindlatel ajahetkedel, mitte pidevalt. Ühe rikastatud neuroni esinemissagedus on määratud võrranditega, mis jäljendavad erinevaid bioloogilisi protsesse, millest kõige olulisem on neuroni membraanipotentsiaal [28]. Võrgud, mis koosnevad rikastatud neuronitest, on võimelised töötleva suhteliselt palju andmeid kasutades väikest arvu rikastatud neuroneid [29]. Kui neuron saavutab teatud potentsiaali, siis see rikastub ja neuroni potentsiaal lähtestatakse [28].

### 3.4 Kapselvõrgud

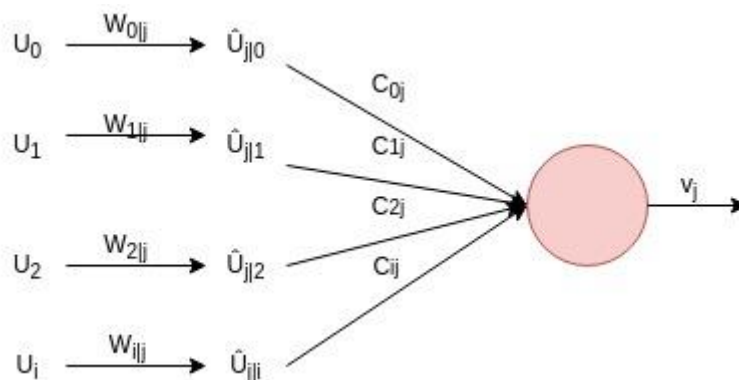
Kapselvõrgud (*Capsule Network*), erinevalt teistest närvivõrkudest, koosnevad kapslitest. Kapsel on neuronite rühm, mille väljundid esindavad sama üksuse erinevaid omadusi [30]. Kapsel õpib tuvastama teatud objekti (näiteks ristkülikut) pildi kindlast piirkonnast ning selle väljundiks on vektor, mille pikkus tähistab hinnangulist tõenäosust, et objekt on pildil olemas ja selle vektori suund kodeerib objekti poosi parameetreid (näiteks täpset positsiooni, rotatsiooni jne) [31]. Ehk siis kui tuvastatud iseärasus liigub pildil ringi või selle olek muutub mingil moel, siis objekti esinemise tõenäosus jääb samaks, kuid selle orientatsioon muutub [19]. See on vajalik, et tulemustes ei tekiks tulemustes vigu ning kapselvõrk ei peaks nägu, millel on silmad, nina ja suu vales järjestuses, enam näoks. Kuna kapselvõrgud tegelevad ruumiliste objektide suhetega, suudavad nad paremini mõista karikatuuride ebaproportsionaalseid

osi (mis eristavadki karikatuure portreedest). See tähendab, et kapselvõrgud klassifitseerimisel täpsemad kui konvolutsioonilised närvivõrgud. Kapsivõrgud vajavad vähem treeningandmeid, et teha sama efektiivseid järeldusi kui konvolutsioonilised närvivõrgud [30].

### 3.4.1 Kapsel

Kapsel on kapselvõrgu väikseim osake. Kapsli toimingud on järgmised:

- sisendvektorite maatriksite korrutamine
- sisendvektorite skalaarkaalude leidmine
- kaalutud sisendvektorite summa leidmine
- kärpimisvektori abil väljundi arvutamine [32].



Joonis 9. Kapsel (autori joonis).

Kapsli sisendvektorid on  $u_0, u_1, \dots, u_i$  ja väljundvektor on  $v_j$ , kaalumatriksid on tähistatud  $W_{0lj}, W_{1lj}, \dots, W_{ijl}$ , ennustusvektorid on tähistatud  $\hat{U}_{jl0}, \hat{U}_{jl1}, \dots, \hat{U}_{jli}$  ja sidestuskoeffitsendid on tähistatud  $c_0, c_1, c_2, \dots, c_n$  (Joonis 9). Kapsli  $j$  väljundvektori saab arvutada kasutades valemit (5) ning kärpimisfunktsiooni (*squash*), kus läheb vaja  $j$  täielikku sisendit  $s_j$  [33].

$$v_j = \frac{\|s_j\|^2}{1 + \|s_j\|^2} \frac{s_j}{\|s_j\|} \quad (5)$$

Kärpimisfunktsiooni eesmärgiks on arvutada kapsli väljundvektori pikkus, mis tähistab hinnangulist tõenäosust selle kohta, kas objekt on antud sisendis olemas [33]. Kapsli väljundvektori pikkus on vahemikus 0 ja 1. Kõikidele kapslitele peale esimese kihi, on

täielikuks sisendiks  $s_j$ , mis leitakse kasutades valemit (6) ning sidestuskoefitsente  $c_{ij}$  ja ennustusvektoreid  $\hat{U}_{ji}$  [33].

$$s_j = \sum_i c_{ij} \hat{u}_{ji} \quad (6)$$

Ennustusvektorid  $\hat{U}_{ji}$  leitakse kasutades valemit (7), kaalumatriksi  $W_{ij}$  ja eelmise kihi kapsli  $i$  väljundi  $u_i$  omavahelisel korrutamisel.

$$\hat{U}_{ji} = w_{ij} u_i \quad (7)$$

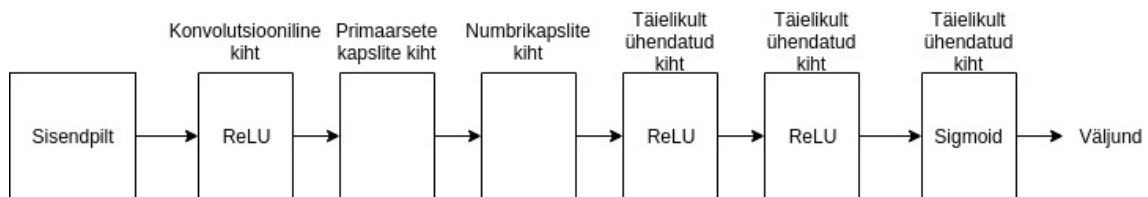
Sidestuskoefitsendid kapsli  $i$  ja kõikide temale eelnenud kihi kapslite vahel annavad summaks 1 ja on kindlaks määratud valemis (8) *softmax* marsruutimisega, mille algsed logitid  $b_{ij}$  on logile eelnenud tõenäosused, et kapsel  $i$  tuleks paari panna kapsliga [33].

$$c_{ij} = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ik})} \quad (8)$$

Logit on kasutusel mis tahes normaliseerimata tõenäosusjaotuse jaoks, logiti väärtus on positiivne kui tõenäosus on suurem kui 0.5, negatiivne kui logiti väärtus on väiksem ning 0 kui väärtus on täpselt 0.5.

### 3.4.2 Kapselvõrgu arhitektuur

Sarnaselt närvivõrkudele koosneb ka kapselvõrk kihtidest. Nendeks on konvolutsiooniline kiht, primaarne kapsel, numbrikapslite kiht ning täielikult ühendatud kihid (Joonis 10).



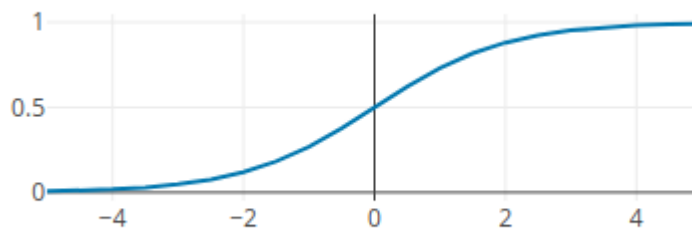
Joonis 10. Kapselvõrgu arhitektuur (autori joonis).

Võrgu sisendiks on värvilised pildid ehk siis kolmedimensioonilised massiivid. Esimeseks kihiks sarnaselt konvolutsioonilisele närvivõrgule on konvolutsiooniline kiht, mille eesmärgiks on tuvastada põhilisi pildi tunnusoone. Selle väljund leitakse valemi (2) ehk mittenegatiivse lineaarfunktsiooni abil. Järgmiseks on primaarsete kapslite kiht, mille ülesandeks on moodustada konvolutsioonilise kihi poolt tuvastatud põhiliste tunnusoonte kombinatsioone [34]. Primaarsete kapslite kihi algus sarnaneb konvolutsioonilisele kihile. Iga kapsel saab sisendiks väikese piirkonna pildist ning

püüab kindlaks teha kindla mustri olemasolu [31]. Primaarse kapslite kihi väljund leitakse kasutades valemit (5) ja karpimisfunktsiooni ning selle kihi väljundid on kaheksadimensioonilised vektorid. Peale seda toimub marsruutimine, millega otsustatakse, missugune informatsioon saadetakse edasi järgmisele kihile [35]. Numbrikapslini jõuavad kuueteistkümnedimensioonilised vektorid. Mida pikem on vektor, seda suurema tõenäosusega asub otsitav objekt pildil. Iga eelneva kihi väljund kaalutakse ning saadetakse täielikult ühendatud kihi neuroni sisendisse. Täielikult ühendatud kihis toimuvad kalkultatsioonid järgmise valemi alusel: parameetrite arv = (sisendite arv + vabaliige) \* neuronite arv kihis [35]. Esimese ja teise täielikult ühendatud kihi väljund arvutatakse mittenegatiivse lineaarfunktsiooni abil, kolmanda kihi väljund arvutatakse valem (8) ehk sigmoidfunktsiooni abil.

$$f(x) = \frac{1}{1+e^{-z}} \quad (9)$$

Graafikul näeks sigmoid funktsioon välja järgmine (Joonis 11):



Joonis 11. Sigmoid funktsioon (autori joonis).

Sigmoidfunktsiooni peamine ülesanne on viia väljundväärtused 0 ja 1 vahele. Negatiivsed numbrid muutuvad nulliks ja suured positiivsed numbrid üheks.

Valem (10) on kapselvõrgu kaofunktsiooni arvutamiseks. Selle arvutamiseks on vaja teada kapsli  $k$  kohta järgmist:

- $T_k$ , mis näitab, kas ennustatud kategooria on tõene.  $T_k$  on kas 0 või 1,
- $m^+ = 0.9$  ja  $m^- = 0.1$ ,
- konstanti  $\lambda$ , mida kasutatakse arvuliseks stabiilsuseks, enamasti  $\lambda = 0.5$ ,
- $v_k$ , mis on kapsli väljundvektor.

$$L_k = T_k \max(0, m^+ - \|v_k\|^2 + \lambda(1 - T_k) \max(0, \|v_k\| - m^-))^2 \quad (10)$$

Täielik kadu on kõikide kapslite kadude summa.

### 3.5 Kasutatud tehnoloogiad

Antud lõputöös on kasutusel Pythoni programmeerimiskeel koos erinevate laiendustega. Python on interpreteeritud, objektorienteeritud, kõrgetasemeline keel koos dünaamilise semantikaga [36]. Python on kergelt mõistetav ja kiirelt õpitav programmeerimiskeel, mille võlu peitub lihtsuses. Pythonis on olemas erinevaid masinõppe teekke, mis sobivad närvivõrkude ülesehituse loomiseks.

#### 3.5.1 Keras

Keras on Pythonis kirjutatud närvivõrkude API. See töötati välja keskendudes kiirete eksperimentide võimaldamisele, et oleks võimalik vähese viivitusega jõuda ideest tulemuseni [37]. Kerase süntaks on lihtsalt mõistetav teades närvivõrkude toimimist ja nende ülesehitust. Seetõttu sobib see hästi ka algajatele. Kerase lehel on olemas ka mitmeid koodinäiteid ning andmekogusid närvivõrkude loomiseks ja testimiseks. Kuna Kerase kasutamine on mugav ja küllaltki hea alguspunkt masinõppega tutvumiseks, siis seda kasutati CNNi implementeerimiseks.

#### 3.5.2 TensorFlow

TensorFlow on teek, mida saab rakendada kõikidele masinõppe algoritmidele, ka närvivõrkudele [38]. TensorFlow pakub Pythoni, C++, Java ja teiste programmeerimiskeelte API'sid. TensorFlow on avatud lähtekoodiga, mis teeb selle hästi hooldatuks ning asjakohastatud teegiks. Antud lõputöös valiti kapselvõrgu implementeerimiseks TensorFlow teek, kuna sellele on võimalik hõlpsasti laiendusi kirjutada.

## 4 Eksperimendid

### 4.1 Andmekogum

Antud lõputöö jaoks koguti kokku 200 portreed ja 200 karikatuuri erinevatest ühiskonnategelastest. Kuigi valdav osa piltidest on erinevatest inimestest, siis USA president Donald John Trumpist (Joonis 1 ja 2) ja ekspresident Barack Hussein Obamast (Joonis 12 ja 13) leiab andmekogust üle 20 pildi mõlemas kategoorias.



Joonis 12. Portree Barack H. Obamast [39].

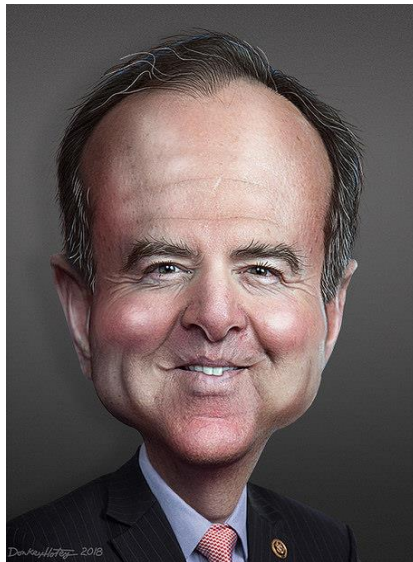


Joonis 13. Karikatuur Barack H. Obamast [12].



## 4.2 Piltide eeltöötlus

Andmekogumis on erinevates suurustes pilte. Kuid närvivõrkude täielikult ühendatud kihid nõuavad kindla suurusega pilte sisenditena. Seetõttu tuleb muuta kõik pildid sama suureks. Enamik portreesid ning karikatuure on algselt ristküliku kujuga, kuid täiesti erinevate mõõtmete suhetega (Joonis 14 ja 15).



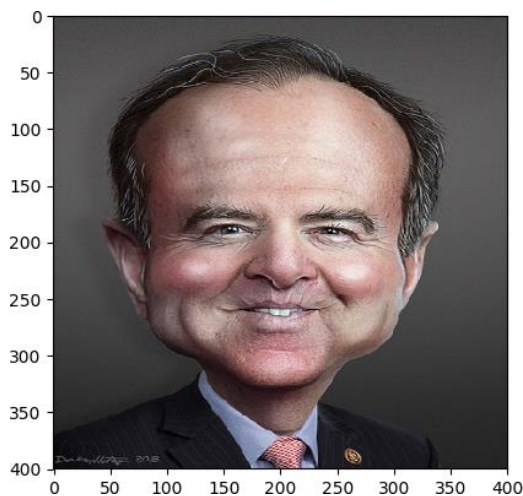
Joonis 14. Katikatuur Adam Schiffist [12].



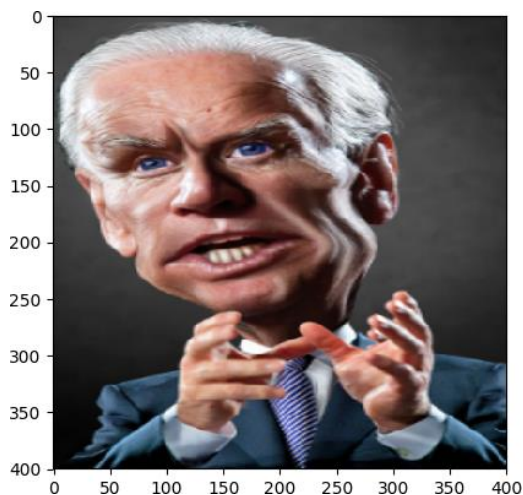
Joonis 15. Karikatuur Joe Bidenist [12].

Kui muuta piltide suurust, tuleb teada nende kõrguste ja laiuste suhteid, mis on tihtipeale erinevad. Algselt katsetati OpenCV teeki sisseehitatud funktsiooni pildi

suuruse muutmiseks (*resize*). Sellele funktsioonile anti kaasa pilt ning laiuse ja kõrguse suurused, näiteks 400 x 400 pikslit (Joonis 16 ja 17).

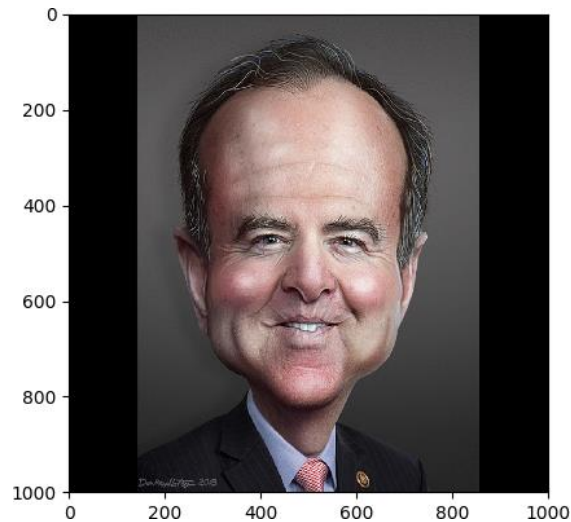


Joonis 16. Ruudukujuline sisendpilt Adam Schiffist [12], autori markeeringuga.

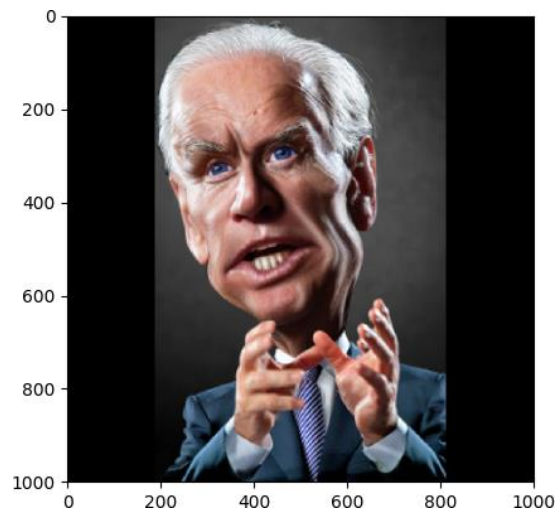


Joonis 17. Ruudukujuline sisendpilt Joe Bidenist [12], autori markeeringuga.

Kuna piltide kõrguse ja laiuse suhted on erinevad, muutuvad igal pildil kujutatud persooni näo ja keha proportsioonid. Portree ja karikatuuri vahe seisneb selles, et mingisuguseid detaile inimese näos ja ilmes tuuakse eriti esile tehes need suuremaks ning väljapaistvamaks. Selline piltide muutmise kaotab tihtipeale pildilt need ebaproportsionaalsed detailid ning konvolutsiooniline närvivõrk ei pruugi enam aru saada, kas tegemist on portree või karikatuuriga. Seetõttu kasutati funktsiooni, mis lisab piltidele mustad ribad äärtesse, et muuta pilt ruudukujuliseks ning seetõttu ei lähe pildilt ka olulised detailid kaotsi (Joonis 18 ja 19).



Joonis 18. Ruudukujuline pilt mustade äärtega Adam Schiffist [12], autori markeeringuga.

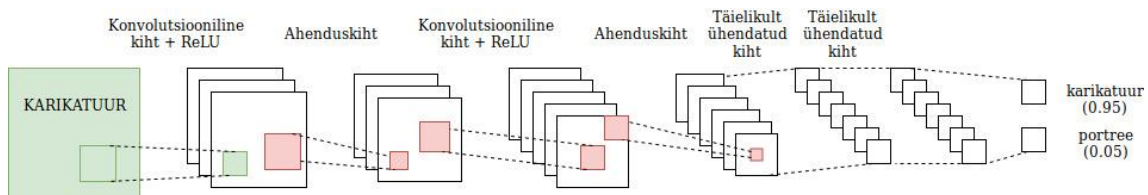


Joonis 19. Ruudukujuline pilt mustade äärtega Joe Bidenist [12], autori markeeringuga.

Eriti ilmekalt tuleb detailide muutumine välja Adam Schiffi karikatuuri puhul, kus algsel pildil on ta pea tehtud ebaloomulikult pikaks, kuid kui kasutada tavapärasest OpenCV pildi mõõtmete muutmise funktsiooni, muutub ta pea lihtsalt ümmarguseks. Kui aga algselt lisada mustad ääred pildile ning seejärel muuta pildi mõõtmeid säilib ka näo ilmekas kuju.

### 4.3 Konvolutsiooniline närvivõrk

Antud lõputöös kasutatav konvolutsiooniline närvivõrk koosneb 7 kihist: sisendkihist, kahest konvolutsioonilisest kihist, kahest ahenduskihist, kahest täielikult ühendatud kihist (Joonis 20).



Joonis 20. Töös kasutatava konvolutsioonilise närvivõrgu arhitektuur (autori joonis).

Närvivõrgu väljundina määratakse pildile kategooria ning näidatakse selle tõenäosust protsentuaalselt. Kõiki konvolutsioonilise närvivõrgu mudeleid treenitakse 75 epohhiga ja minipartiidega suuruses 32. Üks epohh on võrdne kogu treeningandmete ühekordse närvivõrgu läbimisega. Minipartii tähendab, et võetakse üks osa treeningandmetest ühe treeningu iteratsiooni jaoks.

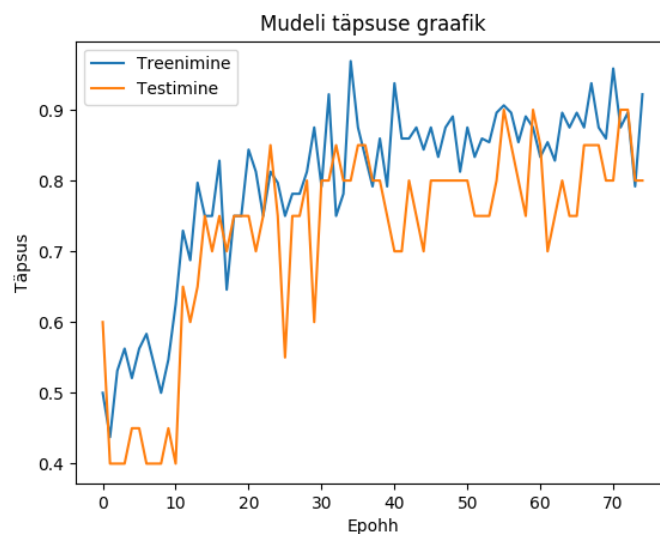
#### 4.3.1 Konvolutsioonilise närvivõrgu treenimine

Konvolutsioonilise närvivõrgu treenimiseks on vaja sisendandmed jagada klasside kaupa kaustadesse. Antud lõputöö puhul on peamine kaust andmekogum, mille sees on eraldi kaustad karikatuur ja portree. Kaustade järgi saab treenimisel konvolutsiooniline närvivõrk teada pildi kategooria. Närvivõrgu käivitamiseks tuleb kaasa anda treeningandmete kaust ning mudeli nimi, mida treenima hakatakse. Närvivõrgu testimiseks tuleb kaasa anda mudeli nimetus ning pilt, mille kategooriat teada soovetakse.

Antud lõputöös treeniti kolme erinevat mudelit: kaks mudelit 50 karikatuuri ja 50 portreega, milles ühe puhul muudetakse piltide laiuse ja kõrguse suhteid, teise puhul jäetakse samaks ning kolmas mudel 200 karikatuuri ja 200 portreega muutmata piltide kõrguse ja laiuse suhteid. Treenimisel jagatakse andmed treeningandmeteks (80%) ja testandmeteks (20%), testandmeid kasutatakse jooksvalt mudeli treenimisel, mis kajastub ka graafikutel testimise joonena.

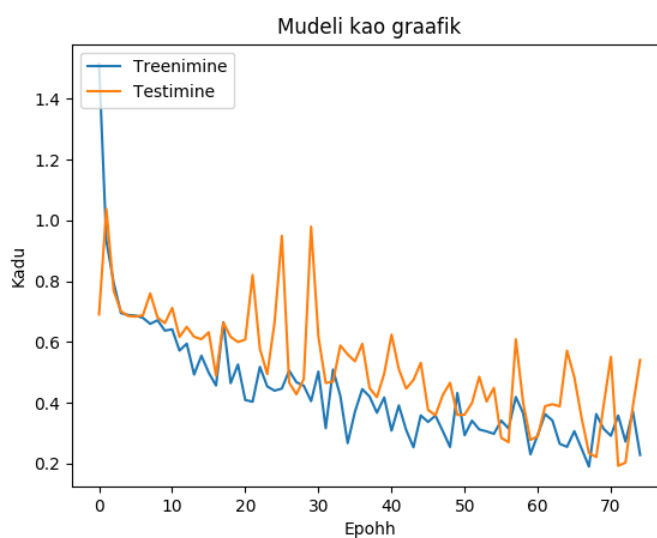
#### 4.3.2 Mudel 50 portree ja 50 karikatuuriga, piltide laiuse ja kõrguse suhted muudetud

Esimesena testiti mudelit, mida treeniti 50 portree ja 50 karikatuuriga. Piltide väiksemaks ja ruudukujuliseks muutmiseks kasutati tavalist OpenCV funktsiooni (*resize*). Sisendpildi suuruseks valiti 32 x 32 pikslit. Testiti 50 pilti ja 50 portreed, mida ei kasutatud treenimiseks. Graafikult on näha, et mudelit treeniti 75 epohhiga ning see on treenimise käigus saavutanud 80% testimise täpsuse (Joonis 21).



Joonis 21. Mudeli täpsuse graafik (autori joonis).

Mudeli kao graafikult on samuti näha, et mudelit treniti 75 epohhiga, kuid mudeli treenimise käigus testimise kadu oli 0.55, mis on väga kõrge (Joonis 22).



Joonis 22. Mudeli kao graafik (autori joonis).

Testimisel suutis mudel klassifitseerida õigesti 47 portreed ning 30 karikatuuri ehk 77% piltidest. Õigesti klassifitseeritud portreede tõenäosusprotsent jäi vahemikku 56 - 100, kusjuures 23 portree tõenäosusprotsent oli 100 (Joonis 23).



Joonis 23. Kõrgeima tõenäosusprotsendiga portree [40], autori markeeringuga.

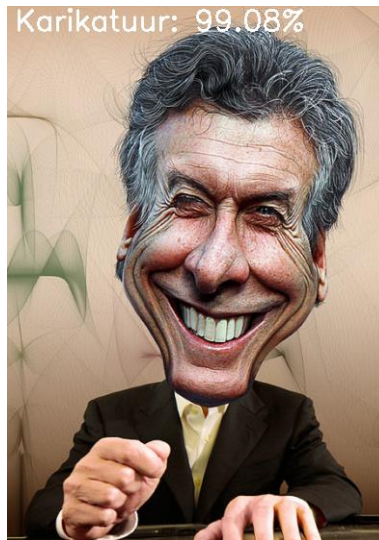
Portreede klassifitseerimisel tegi mudel ka vigu, kuid pigem madalam tõenäosusega vahemikus 50 – 74% (Joonis 24).



Joonis 24. Kõrgeima tõenäosusprotsendiga valesti klassifitseeritud portree [41], autori markeeringuga.

Karikatuuride klassifitseerimisel tegi mudel väga palju vigu suutes õigesse kategooriasse paigutada veidi üle poolte karikatuuridest. Samuti oli õigete klassidega piltide tõenäosus väga varieeruv, vahemikus 50 – 99% (Joonis 25).





Joonis 25. Kõrgeima tõenäosusprotsendiga karikatuur [12], autori markeeringuga.

Valesti klassifitseeritud karikatuuride puhul jäi tõenäosus vahemikku 54 - 100% (Joonis 26).



Joonis 26. Kõrgeima tõenäosusprotsendiga valesti klassifitseeritud karikatuur [12], autori markeeringuga. On näha, et 50 portree ja 50 karikatuuriga treenitud mudel on ebatäpne ning ei tule toime karikatuuride klassifitseerimisega.

#### **4.3.3 Mudel 50 portree ja 50 karikatuuriga, piltide laiuse ja kõrguse suhted muutmata**

Teisena testiti mudelit, mida treeniti samuti 50 portree ja 50 karikatuuriga. Piltide väiksemaks ja ruudukujuliseks muutmisel kasutati funktsiooni, mis lisas piltidele mustad ääred ning seejärel rakendati OpenCV sisseehitatud funktsiooni pildi suuruse muutmiseks. Sisendpildi suuruseks valiti 32 x 32 pikslit. Testimiseks kasutati eelmise

mudeliga sama andmekogu. Antud töö autori hüpoteesiks on, et antud mudel suudab teha täpsemaid järeldusi, kuna piltide laiuse ja kõrguse suhted ei muutu ja seetõttu ei muutu ka märgatavalt piltidel kujutatavate inimeste näo ja keha proportsioonide suhe.

Mudel saavutab treenimisel täpsuse 75% ning kadu jääb 0.5 juurde. Testimisel selgus, et mudel suutis klassifitseerida õigesti 42 portreed ning 38 karikatuuri ehk 80% piltidest, mis on 3% kõrgem kui eelmise mudeli puhul. See tähendab, et jättes piltide kõrguse ja laiuse suhted muutmata kujule, suudab mudel teha täpsemaid järeldusi.

Õigesti klassifitseeritud portreede tõenäosusprotsendi vahemik oli 55 – 100, sealjuures 5 portreed olid 100% tõenäosusega (Joonis 27).



Joonis 27. Kõrgeima tõenäosusprotsendiga portree [42], autori markeeringuga.

Valesti kategoriseeris mudel 8 portreed, tõenäosuse vahemikuga 54 – 93% (Joonis 28).



Joonis 28. Kõrgeima tõenäosusprotsendiga vale kategooriaga portree [43], autori markeeringuga.



Karikatuuride klassifitseerimisel muutus mudel täpsemaks, tulemused jäid vahemikku 50 - 98% (Joonis 29).



Joonis 29. Kõrgeima tõenäosusprotsendiga karikatuur [13], autori markeeringuga.

Valesti klassifitseeritud karikatuuride puhul jäi tõenäosus vahemikku 53 - 98%, kusjuures suurima protsendiga vale klassiga karikatuur oli sama nagu eelmise mudeli puhul, kuid tõenäosus muutus veidi väiksemaks. (Joonis 30).



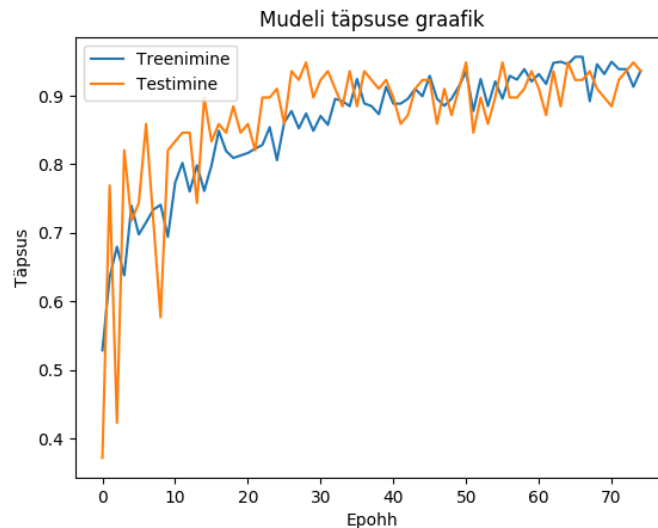
Joonis 30. Kõrgeima tõenäosusprotsendiga vale klassiga karikatuur [13], autori markeeringuga.

#### **4.3.4 Mudel 200 portree ja 200 karikatuuriga, piltide laiuse ja kõrguse suhted muutmata**

Järgmisena testiti mudelit, mida treeniti 200 portree ja 200 karikatuuriga. Piltide väiksemaks ja ruudukujuliseks muutmisel kasutati funktsiooni, mis lisas piltidele mustad ääred ning seejärel lisati OpenCV sisseehitatud funktsioon pildi suuruse

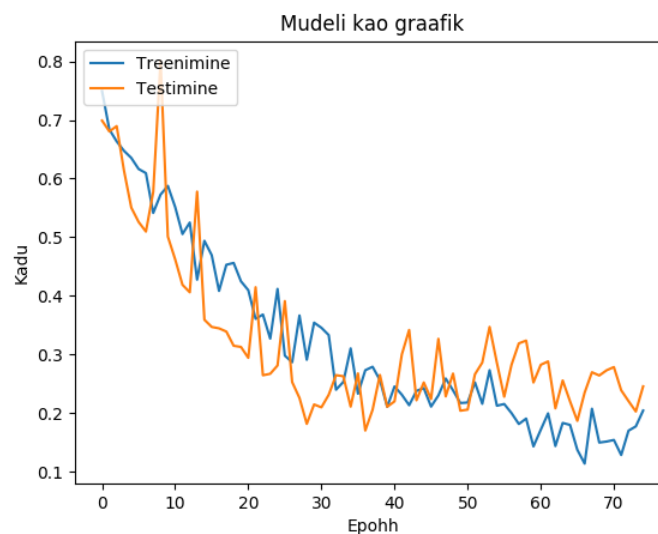
muutmiseks. Sisendpildi suuruseks valiti 32 x 32 pikslit. Piltide kõrguse ja laiuse suhted jäeti samaks, kuna eelmise treenitud mudeli testimisel oli näha, et mudeli tulemused olid täpsemad kui esimese mudeli puhul.

Täpsuse graafikult on näha, et mudelit treeniti 75 epohhiga ning treenimise käigus testimise täpsus jõudis 100% lähedale (Joonis 31).



Joonis 31. Mudeli täpsuse graafik (autori joonis).

Mudeli kadu on väiksem kui eelnevate mudelite puhul, treenimise käigus toimuva testimise kadu on vähenenud 0.25ni (Joonis 32).



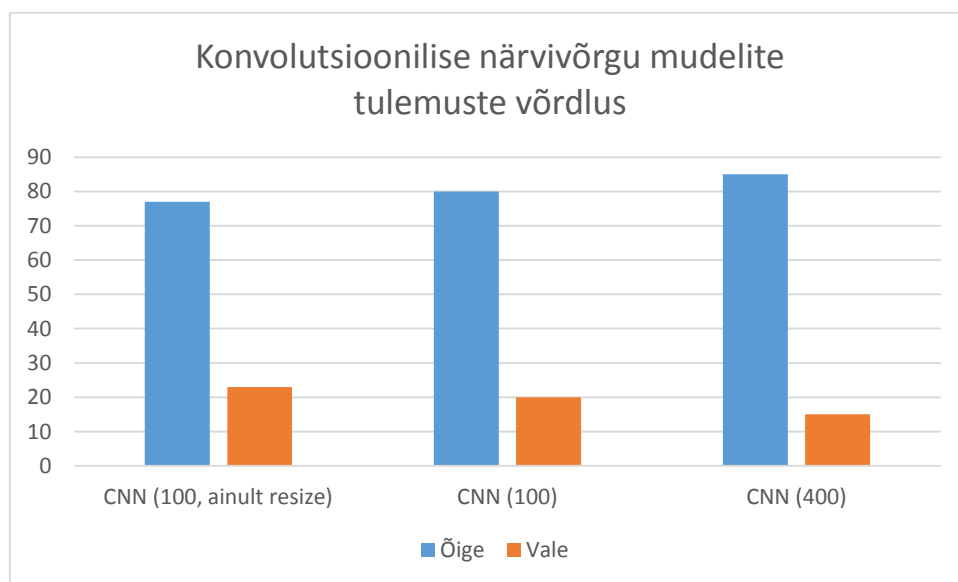
Joonis 32. Mudeli kao graafik (autori joonis).

Mudel suutis testimisel õigestesse klassidesse paigutada 45 portreed ja 40 karikatuuri. Portreede klassifitseerimisel tegi mudel kolm viga vähem ja karikatuuride klassifitseerimisel viis viga vähem kui eelmine mudel, mis näitab, et antud mudel on täpsem kui eelnevad. Mudel klassifitseeris õigesti 85% piltidest, mis näitab, et mida suurem on andmekogu treenimisel, seda täpsemaks mudel muutub.

Portreede puhul, mis olid õigesti klassifitseeritud, jäi tõenäosus vahemikku 56 – 100%. Valesti klassifitseeritud portreede tõenäosus jäi vahemikku 52 – 99%. Karikatuuride puhul jäi õigesti klassifitseeritud piltide tõenäosus vahemikku 52 – 96% ja valesti klassifitseeritud piltide tõenäosus vahemikku 56 – 99%.

#### 4.3.5 Järeldused

Konvolutsioonilise närvivõrgu testimisel oli näha, et väide mida suurem on mudeli treenimisel andmekogu, seda täpsem on mudel, peab paika ka antud ülesande puhul. Graafikult on näha, et mudel 100 pildiga ning muudetud laiuse ja kõrguse suhetega piltidega paigutas õigesti kategooriasse 77% piltidest, samas mudel 400 pildiga ning muutmata kujul laiuse ja kõrguse suhetega piltidega paigutas 85% piltidest õigesti kategooriasse (Joonis 33)..



Joonis 33. Konvolutsioonilise närvivõrgu mudelite tulemuste võrdlus (autori joonis).

Antud parameeter peaks muutuma kapselvõrgu juures vähem oluliseks, aga selle tõestamine või ümber lükkamine toimub järgmises alapeatükis. Seda on näha võrreldes teist ja kolmandat mudelit, mille tõenäosused on vastavalt 80% ja 85%. Suur hulk pilte, mida konvolutsioonilisel närvivõrgul on raske tuvastada, liigitatakse portreede alla.

Portreede ja karikatuuride klassifitseerimise viga oli väga erinev – 400 pildiga treeningandmekoguga mudel määras valesti 5 portreed ja 10 karikatuuri. Paika pidas ka eeldus, et erineva laiuse ja pikkuse suhtega piltide puhul tekib suurem veaprotsent. Kuna närvivõrgu treenimiseks peavad olema kõik pildid sama suurusega, siis lähevad proportsioonid paigast kui kasutada tavalist suuruse muutmise funktsiooni.

Leidus ka pilte, mille klassifitseerisid valesti kõik kolm mudelit (Joonis 24 ja 34).



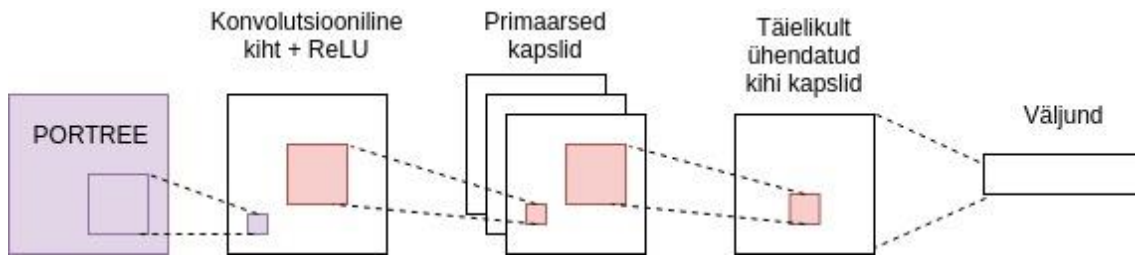
Joonis 34. Karikatuur, mille klassifitseerisid valesti kõik kolm mudelit [13].

Selle põhjuseks võib olla, et treeningandmetes leidus väga sarnaseid pilte teisest kategooriast.

Antud järelduste põhjal valiti edaspidiseks katsetamiseks piltide suuruste vähendamise viis, kus kõigepealt lisati pildile mustad ääred, muutes see ruudukujuliseks ning seejärel vähendati pildi suurust.

#### **4.4 Kapselvõrk**

Antud lõputöös kasutatav kapselvõrk koosneb sisendkihist, konvolutsioonilisest kihist koos mittenegatiivse lineaarfunktsiooniga, kahest kapslikihist ja väljundist (Joonis 35).



Joonis 35. Töös kasutatava kapselvõrgu arhitektuur (autori joonis).

Esimene kapslikiht on primaarne ja teine on täielikult ühendatud kiht. Kapselvõrk kirjutati kasutades TensorFlow teeki juhindudes avalikult kätte saadavast GitHubi koodist [44]. Kuna antud osas ei tegeleta enam analüüsiga kui suur andmekogum sobiks kõige paremini probleemi lahendamiseks ja tõenäosuste protsendid arvutatakse võrkudes erinevalt, siis otsustas töö autor tõenäosusprotsendid antud osast välja jätta.

#### 4.4.1 Kapselvõrgu treenimine

Kapselvõrgu treenimiseks jaotati sisendandmed klasside kaupa kaustadesse. Peamine kaust on andmekogum, mille sees on eraldi portreed ja karikatuurid. Kaustade järgi saab kapselvõrk treenimisel teada pildi kategooria. Kapselvõrgu treenimiseks tuleb käivitada treenimise fail ning kausta nimi, kuhu salvestatakse mudel. Kapselvõrgu treenimisel läbitakse 600 partiid, kusjuures igas partiis on 50 pilti. Antud lõputöös treenitakse kahte erinevat kapselvõrgu mudelit, millest ühe treenimiseks kasutatakse 50 karikatuuri ja 50 portreed ning teise treenimiseks kasutatakse 200 karikatuuri ja 200 portreed.

Piltide suuruste vähendamiseks kasutati nii OpenCV sisseehitatud pildi suuruse muutmise funktsiooni kui ka ise kirjutatud meetodit mustade äärte lisamiseks. Kuna konvolutsioonilise närvivõrgu treenimise ja testimise käigus leiti, et sobivam on kasutada mõlemat funktsiooni piltide väiksemaks tegemisel, ei hakatud kapselvõrgu puhul nende funktsioonide erinevust eraldi testima. Lõputöös eeldatakse, et mida suurem on andmekogum, seda täpsemaks muutuvad kapselvõrgu tulemused ning kapselvõrk suudab väiksema treenitava andmekogumi puhul teha täpsemaid järeldusi kui konvolutsiooniline närvivõrk.

#### 4.4.2 Mudel 50 portree ja 50 karikatuuriga

Esimesena testiti mudelit, mida treeniti 50 portree ja 50 karikatuuriga. Sisendpildi suuruseks valiti 32 x 32 pikslit. Antud töö autor eeldab, et kapselvõrk suudab klassifitseerida täpsemalt pilte kui suutis 50 portree ja 50 karikatuuriga treenitud konvolutsioonilise närvivõrgu mudel.

Kapselvõrk klassifitseeris 84% piltidest õigesti, millest 41 olid portreed ja 43 karikatuurid. Karikatuuri, mille klassifitseeris 100 pildiga treenitud konvolutsioonilise närvivõrgu mudel 63.50% tõenäosusega portreeks ja 400 pildiga treenitud konvolutsioonilise närvivõrgu mudel 91.53% tõenäosusega portreeks, klassifitseeris kapselvõrk karikatuuride alla (Joonis 36).



Joonis 36. Õigesti klassifitseeritud karikatuur [13].

Kuna antud pildi teeb kariaktuuriks see, et ta on ruumiliselt proportsioonidest väljas, ei pruugi konvolutsiooniline närvivõrk sellest aru saada.

#### 4.4.3 Mudel 200 portree ja 200 karikatuuriga

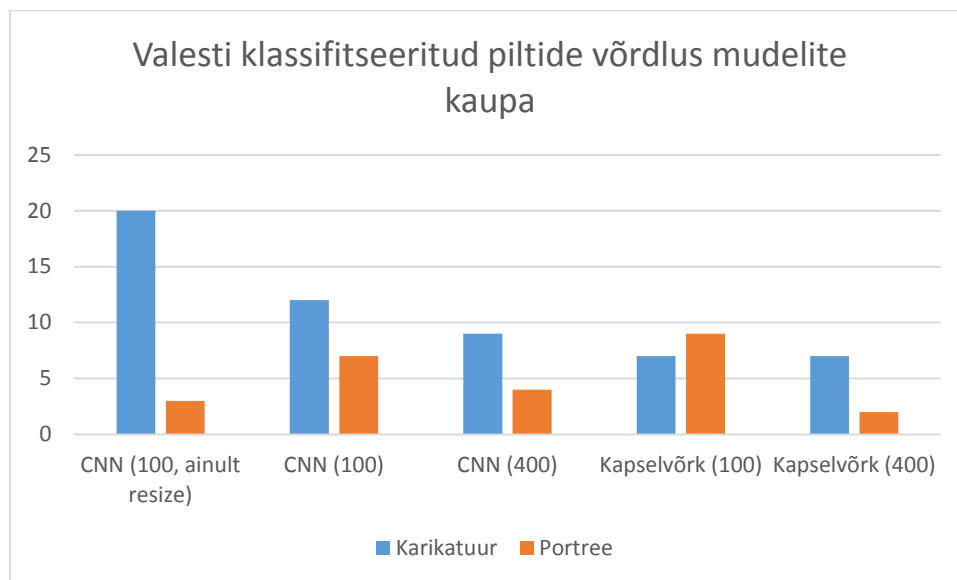
Teisena testiti mudelit, mida treeniti 200 portree ja 200 karikatuuriga. Sisendpildi suuruseks valiti 32 x 32 pikslit. Antud töö autor eeldab, et suurema andmekogumiga treenitav mudel on täpsem kui eelnevad töös välja toodud mudelid.

Kapselvõrk klassifitseeris 91% piltidest õigesti, millest 48 olid portreed ja 43 karikatuurid. See on kõige kõrgem tulemus antud lõputöös ning järelkult pidas eeldus paika – see on kõige täpsem mudel. Antud mudel klassifitseeris õigesti portree joonisel 24, mida ei suutnud ükski eelnevatest mudelitest.

#### 4.4.4 Järeldused

Kapselvõrgud olid palju efektiivsemad eristamaks karikatuure portreedest. See võib tuleneda sellest, et kapselvõrgud oskavad hinnata suhteid pildil olevate objektide vahel ning seejärel teha järeldusi. See omadus tuleb kasuks portreede ja karikatuuride kindlaks määramisel, kuna karikatuurid erinevad portreedest selle poolest, et erinevate näo- ja

kehaosade suhted on portreede mõistes paigast ära. Nii nagu ka konvolutsiooniliste närvivõrkude puhul, pidas kapselvõrkude puhul paika väide, et mida suurem on andmekogum, seda täpsemaid järeldusi suudab mudel teha. Graafikult on näha, et kapselvõrgu tulemused on stabiilsemad, see tähendab, et kapselvõrk suutis ühtlasemalt klassifitseerida pilte ja viga jaotub ühtlasemalt (Joonis 37).



Joonis 37. Valesti klassifitseeritud piltide võrdlus mudelite kaupa (autori joonis).

100 treeningpildi juures klassifitseeris kapselvõrk õigesti 41 portreed ja 43 karikatuuri, samas kui konvolutsiooniline närvivõrk klassifitseeris õigesti 42 portreed ja 38 karikatuuri. Tundub, et konvolutsiooniline närvivõrk on portreede poole kallutatud ehk siis kipub karikatuure portreedeks liigitama. Siit saab järeldada, et kapselvõrku saab visuaalse huumori tuvastamiseks palju rohkem usaldada.

Samas leidis ka pilte, mida ei klassifitseerinud kumbi kapselvõrk õigesti (Joonis 38) ning pilte, mida ei suutnud ükski mudel õigesse kategooriasse liigitada (Joonis 30).



Joonis 38. Karikatuur, mida ei tundnud ära kumbki kapselvõrk [12].



## 5 Kokkuvõte

Töö eesmärgiks oli analüüsida kahe masinõppemeetodi suutlikust huumori tuvastamisel portreede ja karikatuuride näitel. Töö käigus selgus, et mudeli treenimisel tuleb arvestada mitmete erinevate parameetritega. Peamine neist on treeningandmete kogus, mis muudab märgatavalt mudelite täpsust. Lisaks tuleb arvestada ka sellega, et mudel näeb pilti sellisena, millisena see talle ette sööta. Sealjuures peab arvestama, et piltide proportsioonid ei läheks paigast ära, kuna vastasel juhul võib huumor tekkida pildile, kus seda tegelikult vaja ei ole (näiteks muutub nägu piklikuks).

Algoritmide osas tuleb tõdeda, et olenemata sellest, et konvolutsiooniline närvivõrk tuli ülesandega toime, võib siiski pidada kapselvõrku edasiminekuks piltide sisu tuvastamisel. Seda väidet kinnitavad tehtud katsete tulemused, kus kapselvõrk oli märkimisväärselt väiksema sisendandmete juures täpsem. Kui 400 treenitud pildiga konvolutsiooniline närvivõrk tuvastas õigesti 87% piltidest siis kapselvõrk tuvastas 91% piltidest.

Vaadates erinevate närvivõrkude mudeleid, mida treeniti andmekogumiga suuruses 100 pilti, olid tulemused väga erinevad – kui konvolutsiooniline närvivõrk klassifitseeris õigesti 42 portreed ja 38 karikatuuri, siis kapselvõrk klassifitseeris õigesti 41 portreed ja 43 karikatuuri. Kapselvõrgu tulemused olid ühtlasemad ja väiksema veaga. See tähendab, et visuaalse huumori tuvastamisel on kapselvõrgu efektiivsemad, kuna nad suudavad õppida paremini ruumilist paigutust ja proportsioone.

Edasiarendustena võib proovida mõelda piltide muutmist suuremaks, mis omakorda tagaks ka suurema täpsuse. Samuti oleks võimalik kaaluda lisakihtide lisamist konvolutsioonilise närvivõrgu algoritmile ning proovida edasi optimeerida andmete kogust sealjuures arvestades võimaliku ületreenimisega.

## Kasutatud kirjandus

- [1] A. Ziv, "The Social Function of Humor in Interpersonal Relationships," *Soc*, vol. 47, no. 1, pp. 11–18, Jan. 2010.
- [2] A. Reyes, P. Rosso, and D. Buscaldi, "From humor recognition to irony detection: The figurative language of social media," *Data & Knowledge Engineering*, vol. 74, pp. 1–12, Apr. 2012.
- [3] A. Chandrasekaran *et al.*, "We Are Humor Beings: Understanding and Predicting Visual Humor," *arXiv:1512.04407 [cs]*, Dec. 2015.
- [4] Quora, "The Science Behind What Makes Us Laugh," *Forbes*. [Online]. Available: <https://www.forbes.com/sites/quora/2017/04/18/the-science-behind-what-makes-us-laugh/#1b20986d6125>. [Accessed: 18-May-2018].
- [5] A. Karpathy and L. Fei-Fei, "Deep Visual-Semantic Alignments for Generating Image Descriptions," p. 17.
- [6] A. Karpathy, "The state of Computer Vision and AI: we are really, really far away.," *Andrej Karpathy blog*. [Online]. Available: <http://karpathy.github.io/2012/10/22/state-of-computer-vision/>. [Accessed: 18-May-2018].
- [7] D. Yang, A. Lavie, C. Dyer, and E. Hovy, "Humor Recognition and Humor Anchor Extraction," 2015, pp. 2367–2376.
- [8] J. Morreall, "Philosophy of Humor," in *The Stanford Encyclopedia of Philosophy*, Winter 2016., E. N. Zalta, Ed. Metaphysics Research Lab, Stanford University, 2016.
- [9] E. A. Pasquali, "Learning to laugh: humor as therapy," *J Psychosoc Nurs Ment Health Serv*, vol. 28, no. 3, pp. 31–35, Mar. 1990.
- [10] R. Mauro and M. Kubovy, "Caricature and face recognition," *Mem Cognit*, vol. 20, no. 4, pp. 433–440, Jul. 1992.
- [11] S. Craighead, *President Donald Trump poses for his official portrait at The White House, in Washington, D.C., on Friday, October 6, 2017. (Official White House Photo by Shealah Craighead)*. 2017.
- [12] "DonkeyHotey's albums," *Flickr*. [Online]. Available: </photos/donkeyhotey/albums/>. [Accessed: 19-May-2018].
- [13] M. Rouse, "What is AI (artificial intelligence)?," *SearchEnterpriseAi*. [Online]. Available: <https://searchenterpriseai.techtarget.com/definition/AI-Artificial-Intelligence>. [Accessed: 18-May-2018].
- [14] "What is Machine Learning?," *Expert System*, 07-Mar-2017. [Online]. Available: <http://www.expertsystem.com/machine-learning-definition/>. [Accessed: 18-May-2018].
- [15] B. Chandra and R. K. Sharma, "Fast learning in Deep Neural Networks," *Neurocomputing*, vol. 171, pp. 1205–1215, Jan. 2016.
- [16] Y. Guo, Y. Liu, A. Oerlemans, S. Lao, S. Wu, and M. S. Lew, "Deep learning for visual understanding: A review," *Neurocomputing*, vol. 187, pp. 27–48, Apr. 2016.

- [17] S. Agatonovic-Kustrin and R. Beresford, “Basic concepts of artificial neural network (ANN) modeling and its application in pharmaceutical research,” *Journal of Pharmaceutical and Biomedical Analysis*, vol. 22, no. 5, pp. 717–727, Jun. 2000.
- [18] L. Claesson and B. Hansson, “Deep Learning Methods and Applications: Classification of Traffic Signs and Detection of Alzheimer’s Disease from Images.” 2017.
- [19] M. Pechyonkin, “Understanding Hinton’s Capsule Networks. Part I: Intuition.,” *Medium*, 03-Nov-2017. .
- [20] A. Jain, “Fundamentals of Deep Learning - Starting with Artificial Neural Network,” *Analytics Vidhya*, 16-Mar-2016. [Online]. Available: <https://www.analyticsvidhya.com/blog/2016/03/introduction-deep-learning-fundamentals-neural-networks/>. [Accessed: 18-May-2018].
- [21] Aphex34, *Typical CNN architecture*. 2015.
- [22] M. A. Nielsen, “Neural Networks and Deep Learning,” 2015.
- [23] “Convolutional Neural Networks for Visual Recognition.” [Online]. Available: <http://cs231n.github.io/convolutional-networks/>. [Accessed: 18-May-2018].
- [24] A. Saxena, “Convolutional Neural Networks (CNNs): An Illustrated Explanation,” *XRDS*, 29-Jun-2016. [Online]. Available: <https://xrds.acm.org/blog/2016/06/convolutional-neural-networks-cnns-illustrated-explanation/>. [Accessed: 18-May-2018].
- [25] J. Yang, *ReLU and Softmax Activation Functions*. 2018.
- [26] “Loss Functions — ML Cheatsheet documentation.” [Online]. Available: [http://ml-cheatsheet.readthedocs.io/en/latest/loss\\_functions.html](http://ml-cheatsheet.readthedocs.io/en/latest/loss_functions.html). [Accessed: 18-May-2018].
- [27] J. Samuel, “Neurovörgud,” *Medium*. [Online]. Available: <https://medium.com/@Tulevik/neurov%C3%B5rgud-c9b57b3f8673>. [Accessed: 18-May-2018].
- [28] D. Soni, “Spiking Neural Networks, the Next Generation of Machine Learning,” *Towards Data Science*, 11-Jan-2018. [Online]. Available: <https://towardsdatascience.com/spiking-neural-networks-the-next-generation-of-machine-learning-84e167f4eb2b>. [Accessed: 18-May-2018].
- [29] F. Ponulak and A. Kasinski, “Introduction to spiking neural networks: Information processing, learning and applications,” *Acta Neurobiol Exp (Wars)*, vol. 71, no. 4, pp. 409–433, 2011.
- [30] G. Hinton, S. Sabour, and N. Frosst, “Matrix capsules with EM routing,” p. 15, 2018.
- [31] A. Géron, “Introducing capsule networks,” *O’Reilly Media*, 06-Feb-2018. [Online]. Available: <https://www.oreilly.com/ideas/introducing-capsule-networks>. [Accessed: 18-May-2018].
- [32] M. Pechyonkin, “Understanding Hinton’s Capsule Networks. Part II: How Capsules Work.,” *Medium*, 15-Nov-2017. .
- [33] S. Sabour, N. Frosst, and G. E. Hinton, “Dynamic Routing Between Capsules,” *arXiv:1710.09829 [cs]*, Oct. 2017.
- [34] “Understanding Hinton’s Capsule Networks. Part IV: CapsNet Architecture.” [Online]. Available: <https://medium.com/@pechyonkin/part-iv-capsnet-architecture-6a64422f7dce>. [Accessed: 18-May-2018].
- [35] N. Bourdakos, “Understanding Capsule Networks — AI’s Alluring New Architecture,” *freeCodeCamp*, 12-Feb-2018. [Online]. Available: <https://medium.freecodecamp.org/understanding-capsule-networks-ais-alluring-new-architecture-bdb228173ddc>. [Accessed: 18-May-2018].

- [36] “What is Python? Executive Summary,” *Python.org*. [Online]. Available: <https://www.python.org/doc/essays/blurb/>. [Accessed: 18-May-2018].
- [37] “Keras Documentation.” [Online]. Available: <https://keras.io/>. [Accessed: 18-May-2018].
- [38] N. Mahesh, “Understanding a TensorFlow program in simple steps.,” *Towards Data Science*, 26-Sep-2017. [Online]. Available: <https://towardsdatascience.com/understanding-fundamentals-of-tensorflow-program-and-why-it-is-necessary-94cf5b60e255>. [Accessed: 18-May-2018].
- [39] T. O.-B. T. Project, *Poster of Barack Obama's official portrait as president-elect*. 2008.
- [40] G. S. from P. America AZ, United States of, *Cersei Lannister cosplayer at the 2017 Con of Thrones at the Gaylord Opryland Resort & Convention Center in Nashville, Tennessee. Please attribute to Gage Skidmore if used elsewhere*. 2017.
- [41] “Bernie Sanders.jpg,” *Wikimedia Commons*. [Online]. Available: [https://commons.wikimedia.org/wiki/File:Bernie\\_Sanders\\_\(19197909424\)\\_\(cropped\).jpg](https://commons.wikimedia.org/wiki/File:Bernie_Sanders_(19197909424)_(cropped).jpg). [Accessed: 20-May-2018].
- [42] Ståle Grut/NRKbeta, *Jake Tapper*. 2017.
- [43] B. Barata, *Marcela Temer em setembro de 2016*. 2016.
- [44] “A Tensorflow implementation of CapsNet(Capsules Net) apply on german traffic sign dataset.” [Online]. Available: <https://github.com/thibo73800/capsnet-traffic-sign-classifier>. [Accessed: 18-May-2018].

## Lisa 1 – Treeningandmekogum



Treeningandmekogumi portreede näited.



Treeningandmekogumi karikatuuride näited.