27. starptautiskā studentu zinātniski praktiskā konference
Cilvēks.Vide.Tehnoloģijas

KONVOLŪCIJAS NEIRONU TĪKLA APMACĪBA ROKAS ŽESTU ATPAZĪŠANAI PĒC KAGGLE ASL ALPHABET DATU KOPAS

TRAINING OF A CONVOLUTIONAL NEURAL NETWORK FOR HAND GESTURE RECOGNIZING ON THE KAGGLE ASL ALPHABET DATASET

Autors: **Gļebs VITUŠKINS,** e-pasts: glebva2202@gmail.com
Zinātniskā darba vadītājs: **Sergejs KODORS, Dr.sc.ing.,** e-pasts: sergejs.kodors@rta.lv
Rēzeknes Tehnoloģiju akadēmija, Atbrīvošanas aleja 115, Rēzekne

Abstract. *Nowadays, hand gesture recognizing is important topic. It is used in virtual assistant work, for sign language translation, in virtual and augmented reality applications, and in entertainment services. The paper deals with the convolutional neural network training using different technologies. The neural network is trained to classify American manual alphabet and 3 extended gestures using photographs. The open access dataset Kaggle ASL Alphabet was used for training. Kaggle ASL Alphabet provides 87000 images of 29 classes for image classification and hand gesture recognizing.*

Keywords: *neural network, Kaggle, recognizing, sign language, TensorFlow.*

Ievads

Mašīnmācīšanās un datorredzes izmantošana ir mainījusi daudzus mūsdienu tehnoloģiju aspektus, un roku žestu atpazīšana nav izņēmums. Roku žestu atpazīšana plaši izmantota virtuālā asistenta darbā, piemērām atbildēt un beigt tālruņa zvanus, sākt un beigt sarunu ar asistentu utt. [1], ka arī izmantota surdotulkojumā, virtuālās un paplašinātas realitātes lietojumprogrammas.

Lai apmācīt datoru atpazīt rokas žestus, tiek pielietota konvolūcijas neironu tīklu arhitektūra (*convolution neural network*, tālāk *CNN*). CNN plaši izmantota datorredzes. Tos izmanto attēlu klasifikācijai, objektus un rakstus atpazīšanai. Konvolūcijas neironu tīklu pamatideja ir attēlam piemērot filtrus. Filtri pārvietojas pa attēlu un veic konvolūcijas operācijas. Šīs darbības rezultātā tiek iegūts jauns attēls, kur ir informācija par dažādām attēla īpašībām, piemēram, par kontūram, malām un faktūrām.

Šajā rakstā es pētīju iespēju izmantot neironu tīklus, lai atpazīt rokas žestus izmantojot CNN arhitektūru. Spēja atpazīt žestus varētu būt integrētā dažādos lietojumprogrammas, piemērām, integrēt automobiļa multimedijā, lai uzlabotu ceļu satiksmes drošību izmantojot "*hands-free*" risinājumus.

Pētījuma mērķis: izstrādāt konvolūcijas neironu tīklu, kas spēj atpazīt rokas žestus.

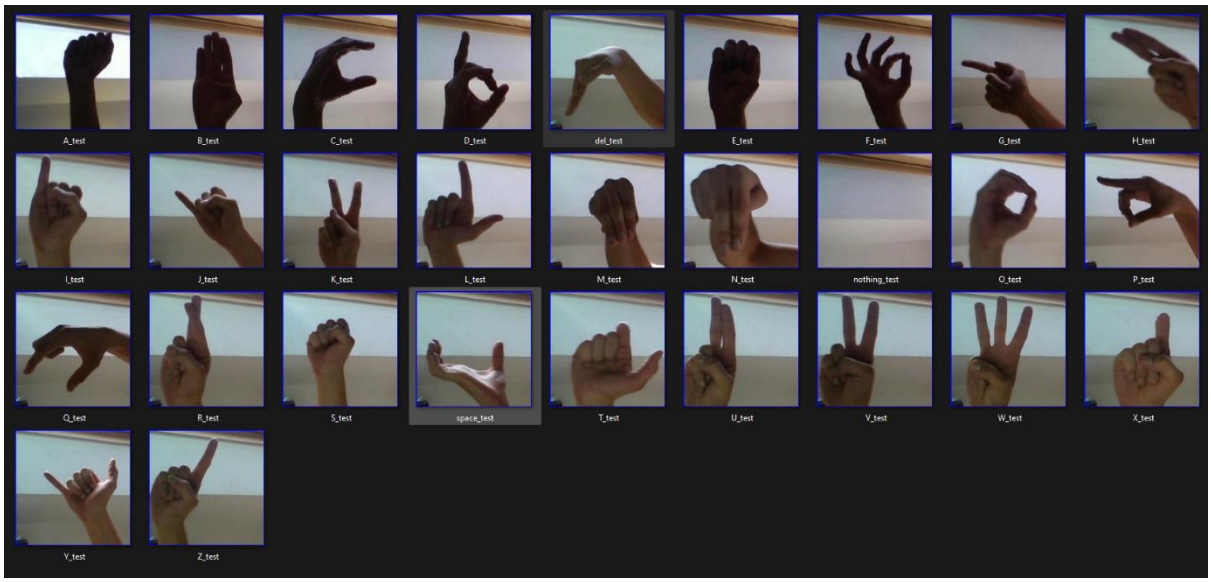
Uzdevumi:

1) Sagatavot datu kopu neironu tīkla apmācīšanai.
2) Izveidot un apmācīt konvolūcijas neironu tīklu.
3) Novērtēt neironu tīklu precizitāti.

Materiāli un metodes

Lai apmācīt neironu tīklu, tiek izmantota Kaggle ASL Alphabet datu kopa (skat. 1.att.). Datu kopa satur 87000 attēlus ar izmēriem 200x200 pikseļus. Attēli ir sadalīti uz 29 klasēm, kur 26 klases ir angļu alfabēts(A-Z) un 3 klases priekš space, dzēst un nekas(*SPACE, DELETE, NOTHING*). Katrā klasē ir 3000 attēlus. Datu kopa ir sadalīta 3 apakškopas: treniņš – 72%, derīgs – 18%, testēšana – 10%. (train – 72%, validation – 18%, test – 10%). Sadalījums nepieciešams, lai pārbaudīt neironu tīklu precizitāte. Visi attēli tiek samazināti līdz 32 uz 32 pikseļa, lai paātrināt modeļu apmācību bez būtiskiem informācijas zudumiem.

https://doi.org/10.17770/het2023.27.7377



1. attēls. Piemēri no Kaggle ASL Alphabet datu kopas

Lai apmācīt neironu tīklu, tiek izmantota Google Colab vidi. Google Colab ir bezmaksas serviss, kur Google nodrošina izveidot un palaist projektus. Google Colab piedāvā mākoņskaitļošanu ar videokartēm(Graphics Processor Unit, tālāk GPU), kas ievērojami samazina neironu tīkla apmācības laiku.

Lai apmācīt konvolūcijas neironu tīklu - jāizveido neironu tīkla modelis. Pēc dažādiem mēģinājumiem tiek izstrādātā konvolūcijas neironu tīkla modelis(sakt. 2. att.). Vispirms, tika pievienots pirmais konvolūcijas slānis ar 128 filtriem, izmēriem 3 uz 3 un aktivācijas funkciju *ReLU*. Otrais slānis ir *MaxPooling*, viņš nepieciešams lai samazināt attēlu izmēru, bet saglabāt svarīgākas pazīmes, manā gadījumā attēls tiek samazināts divreiz. Pēc tam tika pievienots vēl viens konvolūcijas slānis, bet ar 64 filtriem, izmēriem 3 uz 3 un aktivācijas funkciju *ReLU* un vēl viens *MaxPooling* slānis. Piektais slānis ir *Flatten*, kurš pārveido iepriekšējos slāņus izvadi viendimensijas masīvā. Sestais slānis ir *Dense* ar 512 neironiem un aktivācijas funkciju *ReLU*. Un pēdējais slānis ir izejas slānis ar 29 klasēm.

```
model = keras.models.Sequential( [

    keras.layers.Conv2D( 128, (3, 3), activation='relu', name="conv2d_1" ),

    keras.layers.MaxPooling2D( 2, 2 ),

    keras.layers.Conv2D( 64, (3, 3), activation='relu', name="conv2d_2" ),
    keras.layers.MaxPooling2D( 2, 2 ),

    keras.layers.Flatten(),

    keras.layers.Dense(512, activation='relu', name="fcl_1" ),
    keras.layers.Dense(29, activation='softmax', name="out_layer" )
 ] )


model.compile(optimizer= 'adam',
              loss='categorical_crossentropy',
              metrics=['accuracy'])

history = model.fit(
          x_train,
          y_train,
          batch_size = 128,
          epochs = 5,
          validation_split=0.2,
          shuffle = True,
          verbose=1)

model.summary()
print(history)
test_loss, test_acc = model.evaluate(x_test, y_test)
```

2. attēls. Konvolūcijas neironu tīkla modelis

Neironu tikla apmācības parametri(skat 3. att.): apmācības ilgums(*epochs*) = 5, bildes apstrādāšana uzreiz(*batch_size*) = 128, validācijas daudzums(*validation_data*) = 20% no trenēšanas apakškopa un nejauša secība katrā iterācija(*shuffle*).

```
history = model.fit(
          x_train,
          y_train,
          batch_size = 128,
          epochs = 5,
          validation_split=0.2,
          shuffle = True,
          verbose=1)
```

3. attēls. Neironu tikla apmācības parametri**.**

Rezultāti un to izvērtējums

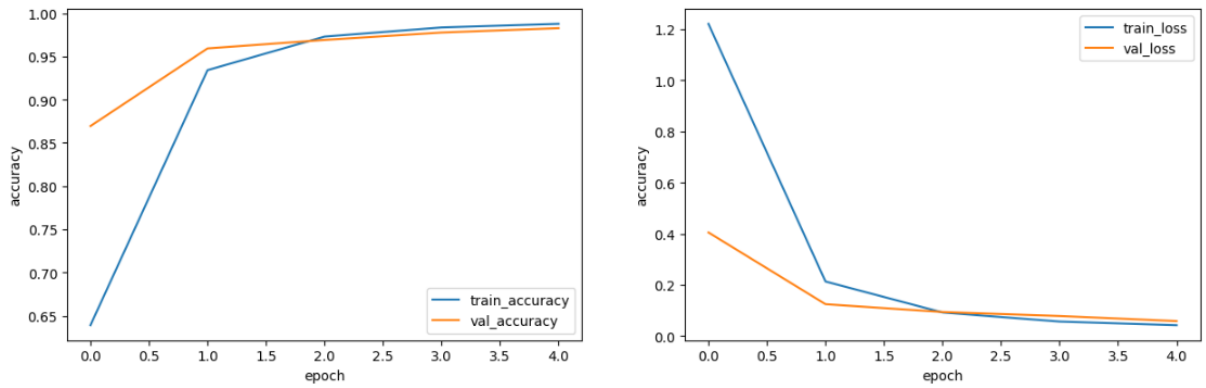
Apmācītā ar *Kaggle ASL Alphabet* datu kopu konvolūcijas neironu tīkls ir parāda šādus rezultātus(skt. 4, 5, 6, att.).

```
Epoch 1/5
490/490 [==============================] - 15s 12ms/step - loss: 1.2211 - accuracy: 0.6390 - val_loss: 0.4053 - val_accuracy: 0.8694
Epoch 2/5
490/490 [==============================] - 5s 11ms/step - loss: 0.2134 - accuracy: 0.9341 - val_loss: 0.1247 - val_accuracy: 0.9592
Epoch 3/5
490/490 [==============================] - 5s 10ms/step - loss: 0.0923 - accuracy: 0.9729 - val_loss: 0.0940 - val_accuracy: 0.9690
Epoch 4/5
490/490 [==============================] - 5s 10ms/step - loss: 0.0567 - accuracy: 0.9835 - val_loss: 0.0785 - val_accuracy: 0.9776
Epoch 5/5
490/490 [==============================] - 5s 11ms/step - loss: 0.0425 - accuracy: 0.9876 - val_loss: 0.0585 - val_accuracy: 0.9826
Model: "sequential"
```

4. attēls. Neironu tīkla rezultāti(apmācības apakškopā)

```
<keras.callbacks.History object at 0x7f43821228b0>
272/272 [==============================] - 1s 3ms/step - loss: 0.0550 - accuracy: 0.9821
```

5. attēls. Neironu tīkla rezultāti(testēšanas apakškopā)



6. attēls. Neironu tīkla precizitātes diagrammas.

Vislielākā neironu tīkla precizitātē pēc apmācības ir 98,76%. Precizitāte uz testēšanas apakškopās ir 98.21%. Tas nozīmē, ka modelis pareizi klasificēja vairāk nekā 98% attēlus. Diagrammās var redzēt, ka precizitāte katra iterācijā pieaug un zaudējumi samazinās. Tās nozīme, kā modelis ir veiksmīgi apmācīta.

Lai uzlabotu neironu tīklu precizitātē, var izmantot sarežģītāku konvolūcijas modeli, izmantot argumentāciju(jaunus bildes izveidošana, pārveidojot esošos attēlus) vai palielināt datu kopas apjomu. Ka arī, neironu tīklu varētu būt pārbaudīta uz saviem datiem, lai novērtēt

tīklu precizitātē dažādas situācijas, piemēram, pagriezt, pārvietot, pievienot troksni, palielināt vai samazināt spilgtumu.

Secinājumi

Izmantojot konvolūcijas neironu tīklu arhitektūru un *Google Colab* vidi, tika apmācīts neironu tīkls, kas atpazīst 29 rokas žestus ar precizitāte aptuveni 98%(uz apmācības un testēšanas apakškopas). Apmācībām tika izmantota datu kopa *Kaggle ASL Alphabet,* kas sastāv no 87000 attēliem. Objektīvāku rezultātu iegūšanai var izmantot sarežģītāku konvolūcijas modeli, izmantot argumentāciju vai palielināt datu kopas apjomu.

Summary

Nowadays, hand gesture recognition is used in various spheres such as virtual assistant work, sign language translation, virtual and augmented reality, and entertainment services. Convolutional neural networks (CNN) are commonly used in computer vision for image classification, object detection, and pattern recognition. In this study, the feasibility of using neural networks with CNN architecture for hand gesture recognition was investigated.

The neural network was trained using the Kaggle ASL Alphabet dataset, which consists of 87,000 images with dimensions of 200x200 pixels. The images are categorized into 29 classes, including 26 English alphabet classes (A-Z) and 3 additional classes for space, delete, and nothing. The dataset was split into three subsets: training – 72%, validation – 18%, and testing – 10%. All images have been resized to 32 by 32 pixels to accelerate model training without significant loss of information.

The neural network achieved an accuracy of approximately 98% on both the training and testing subsets for recognizing 29 hand gestures using the CNN architecture and the Google Colab environment. To obtain more objective results, a more complex convolutional model could be used, augmentation could be applied, or the dataset size could be increased.

Literatūra

1. Google, Use gestures to control your Google Assistant on headphones https://support.google.com/assistant/answer/7513985?hl=en&co=GENIE.Platform%3DAndroid
2. Sumit Saha, A Comprehensive Guide to Convolutional Neural Networks — the ELI5 way. https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53
3. Benjamin Zeman, What is Google Colab? https://www.androidpolice.com/google-colab-explainer/
4. TensorFlow. https://www.tensorflow.org/tutorials
5. Kiprono Elijah Koech, The Basics of Neural Networks (Neural Network Series). https://towardsdatascience.com/the-basics-of-neural-networks-neural-network-series-part-1-4419e343b2b