Estonian Fingerspelling Recognition

Hendrik Matvejev, **Priit Peterson**, **Karl Kristjan Puusepp**

University of Tartu

Introduction

The Estonian fingerspelling alphabet consists of 32 different hand gestures, each representing a letter from the Estonian alphabet [1]. These gestures are used for spelling out names and other proper nouns by the Estonian deaf and hard-of-hearing community. While there exist models for hand gesture detection from video, none are tailored for detecting Estonian fingerspelling gestures. Therefore, we sought out to create a machine learning model for this task. The model is based on Google's Mediapipe hand gesture recogniser [2] which detects hand landmarks (coordinates of 21 hand key points) from images and then uses them to predict a gesture. Additionally, we created a website that uses this model to help Estonian sign language learners practice their fingerspelling in a similar fashion to touch typing websites.

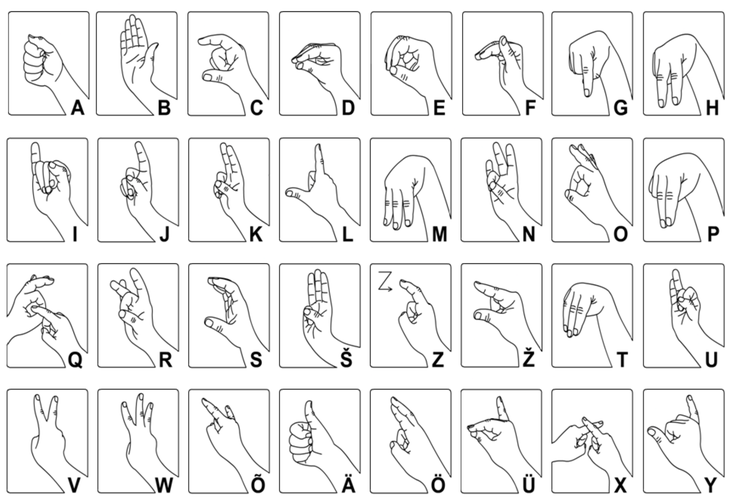

Fig 1. Estonian fingerspelling gestures [1].

Data gathering

There did not exist a suitable dataset for creating the model, hence we had to create it ourselves. We gathered data by photographing eight individuals (four men, four women – aged from 18 to 21 years old) in various locations, capturing over 200 images for each sign. The images were captured manually using a laptop webcam, which is in line with the final use case of the product, as the model is meant to be run through a live webcam feed. Participants were asked to slightly vary their signs to help the model learn both the essence of a sign and its spatial properties. This intentional diversity ensures the model's effectiveness in recognising Estonian fingerspelling signs across different perspectives and real-world scenarios. The images were then pruned of any metadata and renamed.

Hand landmark extraction and data validation

The model is not trained with the images directly, but with 21 hand key points called landmarks. This ensures that the key features of the data are represented in the simplest way possible and also makes training the model feasible. The hand landmarks are generated using Google's Mediapipe hand landmark detection model [3] which takes in images and outputs the coordinates of the key points in 3-dimensional space.

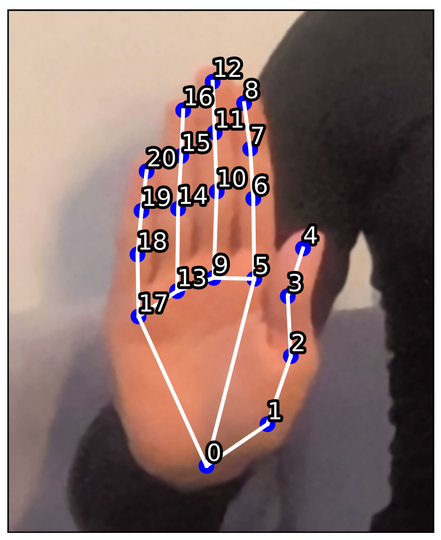

Fig 2. Hand landmarks on cropped image of B.

We generated cropped images with hand landmarks and connectors between them to check whether the landmarks were detected correctly. The images, where the landmarks were detected incorrectly, the gesture was wrong, or there were no landmarks detected, were removed manually. We also separated one-handed and two-handed gestures and opted to leave out the two-handed gestures ("X" and "Q"), as these needed additional work. Since the gathered dataset had an unequal number of right-handed and left-handed poses, the images had to be copied and mirrored before landmark extraction to account for both left- and right-handed gestures, effectively doubling the size of the dataset.

Model training and evaluation

Our model is based on the Mediapipe hand gesture recogniser which itself is a custom Tensorflow model with some additional features for hand gesture recognition training. Due to the highly specialised nature of the model, the training time was quite fast and required very few epochs to reach acceptable accuracy and loss values. For training, the data was split into chunks for training, validation and model evaluation with each subset making up 80, 10 and 10 percent respectively.
A simplified grid search was performed to determine the best values for the dropout rate, batch size, amount and size of hidden layers and the number of epochs. To combat overfitting we chose candidate models that had an accuracy between 75 and 90 percent favouring ones with a fewer number of epochs. These were then tested in the final use case to find the model that performed best in real-world scenarios. The final model has accuracy and loss values of 0.85 and 0.22 respectively.

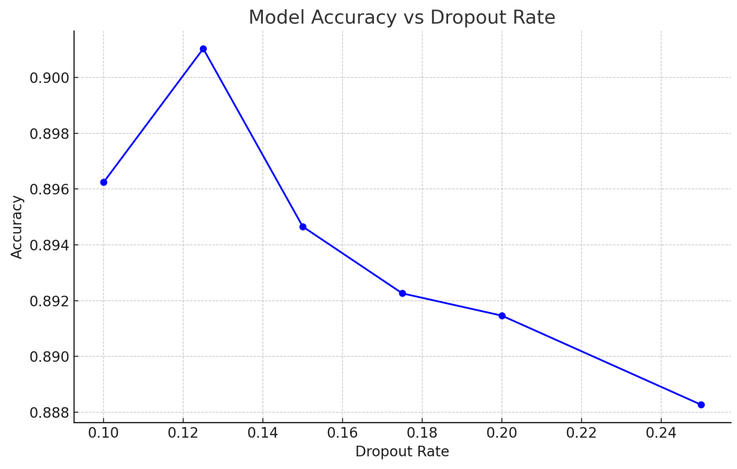

Fig 3. Model accuracy versus dropout rate.

Analysis

Performing principal component analysis (PCA) on the set of extracted landmarks revealed some regularities in the data. For example – letters with two distinctly different gestures (such as "H", that has two related hand gestures, one from the back and another from the front) seemed to form two separate, although more spread-out clusters. Additionally – the first principal component has an explained variance of 25% with the top three having a composite explained variance of 48%. Plotting the top three principal components revealed a larger clustering of gestures where the fingers tend to point up and a smaller cluster of "downward-facing" gestures separated mainly on the axis correlated to PC1. Upon further inspection PC1 seems to correlate well with the number of fingers facing up or down.

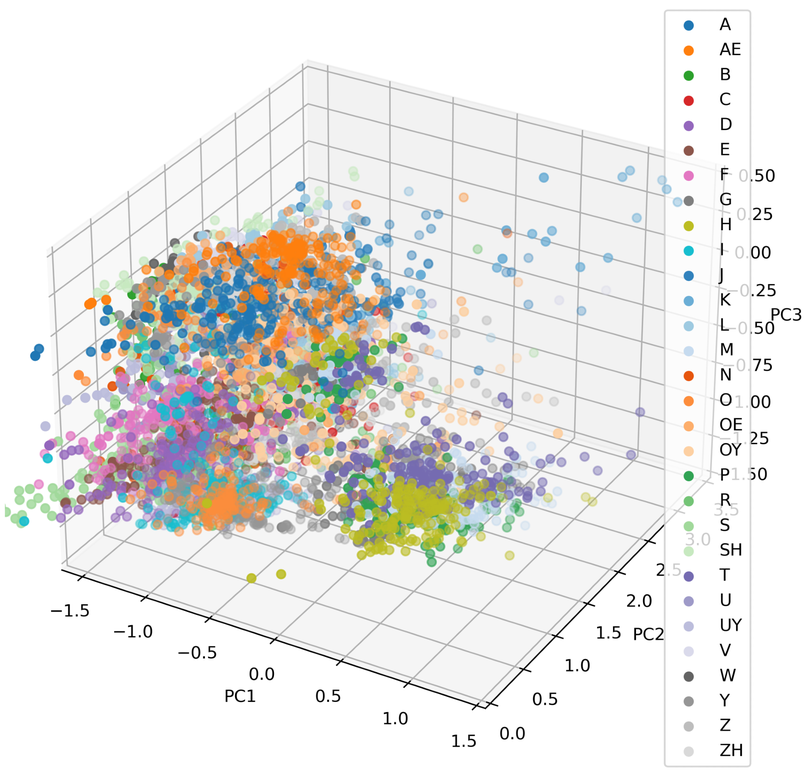

Fig 4. PCA components by sign.

References

[1] . *Eesti viipekeele sõnastik*. [Online; accessed 2023-12-10].
[2] . "Gesture recognition task guide". In: *Google for Developers* (). [Online; accessed 2023-12-10].
[3] . "Hand landmarks detection guide". In: *Google for Developers* (). [Online; accessed 2023-12-10].