

Handshape Classification in a Reverse Dictionary of Sign Languages for the Deaf

Alikhan Abutalipov¹, Antonio Cerone¹[0000-0003-2691-5279], Aigerim Janaliyeva¹, and Anara Sandygulova²

¹ Department of Computer Science, Nazarbayev University, Nur-Sultan, Kazakhstan
alikhana.abutalipov@nu.edu.kz, antonio.cerone@nu.edu.kz,

aigerim.janaliyeva@nu.edu.kz

² Department of Robotics and Mechatronics, Nazarbayev University, Nur-Sultan, Kazakhstan

anara.sandygulova@nu.edu.kz

Abstract. This presentation showcases the work that aims to build a user-friendly mobile application of a reverse dictionary to translate sign languages to spoken languages. The concept behind the reverse dictionary is the ability to perform a search by demonstrating a handshape in front of a mobile phone's camera. The user would be able to use this feature in two ways. Firstly, the user would be able to search for a word by showing a handshape for the application to provide a list of signs that contain that handshape. Secondly, the user could fingerspell the word letter by letter in front of the camera for the application to return the sign that corresponds to that word. The user can then look through the suggested videos and see their written translations. To offer other functionalities, the application also has Search by Category and Search by Word options. Currently, the reverse dictionary supports translations from Russian Sign Language (RSL) to Russian.

Keywords: reverse dictionary, human-computer interaction, iOS application, Russian Sign Language (RSL)

1 Introduction

Deaf communities around the world use sign languages for everyday communication. Each country or region has its own sign language. Contrary to popular belief, Russian Sign Language (RSL) does not share structure or grammar with the Russian language. In addition, people native to RSL do not necessarily know how to read and write Russian and have to learn it as a foreign language.

Most online sign language (SL) dictionaries are alphabet-based which are convenient for people that are fluent in written languages. When searching for a sign, they need to know the written translation of it and search by its first letter. However, such functionality is useful for people who want to learn SL and cannot provide a reverse option - searching for meaning of unfamiliar signs.

There exists only a few reserve dictionaries where searching by sign is performed by one of its components, such as handshapes. Nonetheless, it is still not

user-friendly as each handshape is described in a written form. Usually these descriptions are compiled by professional SL linguists, which makes it hard for a non-expert user to understand the description. Sometimes the pictorial representations of the handshapes are provided too, but then creation of such dictionaries for every sign is time-consuming.

Therefore, this work aims to build an automatic reverse dictionary where a search is performed in the most natural way - searching by demonstration. Since each sign in a sign language consists of one or several handshapes, searching by handshape demonstration would yield the most intuitive method for people native to sign languages.

2 System Design

First, we created a database where each sign video has a list of handshapes corresponding to that sign. We used a publicly available dictionary of RSL from the Spread the Sign dictionary ³. Thus, every frame of the sign video was cropped to contain only the hand region using “Hand-CNN” pre-trained hand detection model [3]. Then, we utilized “Deep Hand” pre-trained handshape recognition model [1] to classify handshapes in each sign video. Once the database was ready, we built a system consisting of two main components, an iOS mobile application and a server that runs “Hand-CNN” and “Deep Hand” models. When a user takes a photo of a handshape, the application would send the image over HTTP request to the server, which in turn would classify the handshape in the photo and return the result to the application via an HTTP request. The application then would show the user the signs that contain the user’s handshape by searching the database that we created previously. When a user takes a photo of another handshape, the just-described process repeats, but this time the application shows the signs that contain both handshapes. The more handshapes are shown, the narrower the search is. In order to adapt the handshape classification to support RSL and fingerspelling in RSL, we utilized manually labeled dataset of RSL handshapes [2] as well as previously collected Cyrillic fingerspelling dataset [4] to perform transfer learning of “Deep Hand” model to make two models for RSL handshape recognition and Cyrillic fingerspelling. In the end, the number of classes was 29 for 33 letters in the Russian alphabet as some cases were combined due to being different only in the movement, for example, signs for the letters И, Ы, III, III. For the transfer learning, we decreased the overall learning rate from 0.0005 to 0.0002 while increasing the learning rate of the final layer by 2 and used the RSL datasets [2,4] to re-train the pre-trained “Deep Hand” model’s weights. The results were: Top-1 results refer to the out deemed most probable by a model, while Top-5 results refer to the 5 most probable outputs of models. The reason for transfer learning was two-folds: first, the “Deep Hand” model already showed rather good results on their dataset: 85% for Top-1 results and 94.8% for Top-5 results [1] (see Table 1). It was beneficial to use the model’s “knowledge”. Secondly, the size of our datasets used for transfer learning was

³ www.spreadthesign.com

much smaller than the dataset used to train “Deep Hand” in [1]: 3201 images for 36 classes of the Handshapes model and 1587 images for the Fingerspelling model versus over 1 million handshape images in [1].

Model	Class Number	Top-1 Accuracy [%]	Top-5 Accuracy [%]
Deep Hand [1]	45	85	94.8
Fingerspelling	29	88	97
Handshapes	35	74	94.6

Table 1. Transfer-learned models’ accuracy

2.1 Search by Handshapes

The main functionality of the application is the ability to search for signs by the handshapes that are used to form them. The “Handshape” option from the home view launches the camera view for the user to take a photo of their hand. After taking a photo the “Search by Handshape” view is shown where a top one-third part of the view shows the photos of the handshapes that the user uses to search for a sign. The rest of the view shows the list of signs that contain the user-provided handshapes. The signs are shown as videos in the loop. We assume that because deaf people are proficient in recognizing signs, they will not be confused by simultaneously playing videos of different signs.

The recently taken photo is added to the top part of the view. If the application successfully classifies the handshape, the border around the photo of the handshape turns green. However, if the handshape is not classified or the application cannot reach the server, the image disappears. The user can also add other handshapes. To do so, the user taps on the “camera” button in the top right corner of the view, which presents the camera view, where the user can take a photo of another handshape. Moreover, the user can delete a handshape from the search by long pressing on the photo of the handshape and tapping on the “delete” button that will be shown as the result. The list of signs updates every time a new handshape is added or an existing one is deleted to reflect the most current state of the search. Finally, the user can tap on a sign, which will result in the “Sign” view to be shown.

2.2 Search by Fingerspelling

The other feature of the application is “Search by Fingerspelling”. Similarly to the “Search by Handshapes” it sends a handshape image shown during fingerspelling to the server, which returns back the bounding boxes that bound the hands in the image. The application classifies the image using the locally run “Fingerspelling” model. Here, however, the distinction between these two features is evident. The application does not search for signs immediately, but rather sends another image of handshape to the server and waits for the bounding boxes coordinates. It does

so for a few dozen images, after which it checks, whether there is a particular sign that corresponds to a minimum 80% percent of classified images of handshapes. If so, the application builds a word by adding the letter that is represented by the handshape that reached the 80% threshold. If the threshold is not met, the application discards the oldest frame and sends the latest frame to the server again, and tests for the threshold again. After the word is built, the application sends a query to the database, fetches signs that relate to the built word and shows the result to the user.

2.3 Search by Words

Here, the user sees the list of all words and phrases that the application has in its vocabulary. The user can use the search bar at the top of the view to search for the word or phrase that they want the sign translation for. After the user taps on a specific word or phrase, the video of the signs that correspond to the selected word or phrase is shown in a loop. In addition to that, the word or phrase is shown at the bottom of the screen. Moreover, the user can tap on the star image to mark or unmark the sign as favorite. All favorite signs can be accessed quickly by clicking on the “Favorites” option in the home view. This method of searching will be mostly useful for the people who are learning the sign language. However, deaf people might also find this method useful, as it would allow them to translate unknown Russian words that they encounter.

2.4 Search by Categories

In the home view, when the user taps on the “Category” option, “Search by category” has multiple categories are presented in a way similar to the “search by words” view. By tapping a word or phrase in this list, the list of sign videos are presented.

References

1. O. Koller, H. Ney, and R. Bowden. Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3793–3802, 2016.
2. M. Mukushev, A. Imashev, V. Kimmelman, and A. Sandygulova. Automatic classification of handshapes in Russian Sign Language. In *Proceedings of the LREC2020 9th Workshop on the Representation and Processing of Sign Languages: Sign Language Resources in the Service of the Language Community, Technological Challenges and Application Perspectives*, pages 165–170, Marseille, France, May 2020. European Language Resources Association (ELRA).
3. S. Narasimhaswamy, Z. Wei, Y. Wang, J. Zhang, and M. Hoai. Contextual attention for hand detection in the wild. 2019.
4. N. Tazhigaliyeva, N. Kalidolda, A. Imashev, S. Islam, K. Aitpayev, G. I. Parisi, and A. Sandygulova. Cyrillic manual alphabet recognition in rgb and rgb-d data for sign language interpreting robotic system (slirs). In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4531–4536. IEEE, 2017.