Technologia i Automatyzacja Montażu

Volume 123, Issue 1/2024, Pages 10-15
https://doi.org/10.7862/tiam.2024.1.2

Original Research

SIGN LANGUAGE CLASSIFIER BASED ON MACHINE LEARNING

KLASYFIKATOR JĘZYKA MIGOWEGO OPARTY NA UCZENIU MASZYNOWYM

Camelia AVRAM[1,*], Laura-Flavia PĂCURAR[1], Dan RADU[1]

[1] Department of Automation, Technical University of Cluj Napoca, Ge. Barițiu 26-28, Cluj-Napoca, Romania

* Corresponding author: camelia.avram@aut.utcluj.ro, tel.: 0040 264 401 822

Abstract

Sign language represents an efficient way for individuals with hearing impairments to communicate. We propose a sign recognition system into which several tools are integrated to help with the image pre-processing part. By doing so, a machine learning model was developed that does not require a lot of processing power because instead of using the images themselves, it uses extracted data from them to connect this model to a mobile interface that the users will use to recognise signed letters successfully. The communication between the client and the model is sustained through a local server. Introducing sign language into assembly processes is not only a gesture of respect for diversity and inclusion but also a strategic decision that brings tangible benefits. It improves communication, safety, employee morale and overall efficiency, an essential element in achieving operational excellence and an integrated workplace.

Keywords: sign language in assembly process, sign language classifier, machine learning, mobile application

Streszczenie

Język migowy stanowi skuteczny sposób komunikacji dla osób z upośledzeniem słuchu. Proponujemy system rozpoznawania znaków, w którym zintegrowano kilka narzędzi pomagających we wstępnym przetwarzaniu obrazu. W ten sposób opracowano model uczenia maszynowego, który nie wymaga dużej mocy obliczeniowej, ponieważ zamiast korzystać z samych obrazów, wykorzystuje wyodrębnione z nich dane, aby połączyć ten model z interfejsem mobilnym, którego użytkownicy będą używać do skutecznego rozpoznawania podpisanych liter. Komunikacja między klientem a modelem odbywa się za pośrednictwem lokalnego serwera. Wprowadzenie języka migowego do procesów montażowych to nie tylko gest szacunku dla różnorodności i integracji, ale także strategiczna decyzja, która przynosi wymierne korzyści. Poprawia komunikację, bezpieczeństwo, morale pracowników i ogólną wydajność, co jest istotnym elementem w osiąganiu doskonałości operacyjnej i zintegrowanego miejsca pracy.

Słowa kluczowe: język migowy w procesie montażu, klasyfikator języka migowego, uczenie maszynowe, aplikacja mobilna

1. Introduction

Sign language represents an efficient way for individuals with hearing impairments to communicate. A common mistake is to think that the sign language is universal, but it isn't. Deaf people around the world use different sign languages. These gestures, called signs, represent a linguistically structured system of movements and symbols. Three elements make a gestured symbol a sign: the hand's position, shape, and motion. The most used sign language is ASL – short for American Sign Language. (Alliance, 2024).

It's natural to use your hands when speaking to a deaf or hard-of-hearing person. Even if you understand sign language or have a basic grasp of fingerspelling and try to communicate, the impaired person might have trouble understanding you despite your best intentions.

Luckily, technology's ongoing growth and development have created new opportunities for people with hearing loss to bridge the communication barrier. Also, mobile applications have become quite popular in the last few years due to their ability to simplify various aspects of our daily lives. By combining these two, we get the primary motivation behind this paper: provide a user-friendly tool that can easily recognise and interpret sign language to help people both within the deaf community and those who are interested in learning American Sign Language (ASL).

Sign language plays a key role in assembly processes, not only providing a means to overcome communication barriers for deaf and hard-of-hearing people but also helping to improve overall efficiency, safety, and team morale. In an environment where precision and cooperation are essential and where noise often impedes traditional verbal communication, sign language enables the smooth and rapid exchange of information, which is vital to maintaining work continuity and preventing errors.

Safety is a priority in any manufacturing facility, and effective sign language communication allows for immediate warning of potential hazards and response to emergency situations. This significantly reduces the risk of accidents and provides a safer working environment for all employees.

Integrating sign language into the assembly process also has a positive impact on the workplace atmosphere. It promotes a sense of belonging and equality among employees, regardless of their hearing ability. Employees who feel included and valued are more engaged and productive, which translates into better quality work and operational efficiency.

Providing access to training and assembly instructions in sign language is essential to enable deaf and hard of hearing people to participate fully in production processes. This not only makes it easier for them to understand procedures and standards, but also enables them to develop their skills and qualifications, benefiting both employees and the organisation.

Companies that actively integrate sign language into their assembly processes demonstrate greater innovation and flexibility. By embracing diversity and promoting inclusion, they can benefit from the unique perspectives of their employees, often leading to the identification of new solutions and process improvements.

The paper will be structured as follows: The first section, State of the Art, contains a deep analysis of the subject based on theoretical papers and different approaches to solving the problem. The second section, Machine Learning Model, explains the proposed solution from a theoretical point of view, plus the tools and technologies used. The third section, Detailed Design and Implementation, describes the application's development. The Conclusions section contains a paper synthesis and a critical analysis of the results.

2. State of the art

Sign language recognition systems have seen significant advancements in recent years, driven by computer vision, machine learning, and signal processing developments. These systems aim to facilitate communication for individuals with hearing impairments by automatically interpreting sign language gestures.

According to an article by ALTA Language Services, over 300 sign languages are worldwide. About 70 million people worldwide are hearing-impaired and use sign language to communicate.

Like spoken languages, every country has its way of signing, and each might have different dialects (Alta, 2024). Table 1 presents the primary sign languages worldwide (according to Earthweb 2024).

Table 1. Major Sign Languages around the world

Country/ continent	**Sign Language**	**Abbrev.**
United Kingdom	British Sign Language	BSL
United States of America	American Sign Language	ASL
Australia	Australian Sign Language	Auslan
Japan	Japanese Sign Language	JSL
People's Republic of China	Chinese Sign Language	CSL
Taiwan	Taiwanese Sign Language	TSL
Middle-East	Arabic Sign Language	ArSL
The Islamic Republic of Iran and other Gulf countries	Persian Sign Language	PSL
India	Indian Sign Language	ISL
Vietnam	Vietnam Sign Language	VSL
Ukraine	Ukrainian Sign Language	UKL
Sri Lanka	Sri Lankan Sign Language	SLTSL
Federative Republic of Brazil	Brazilian Sign Language	Libras
Poland	Polish Sign Language	PJM
The Netherlands	Sign Language of the Netherlands	SLN

(Chai, 2013) Presents how hand gesture recognition influences the sign language recognition system. It focuses on previously used classification methods, compares them, and even suggests the most promising ones for future development.

To fully implement a Sign Language Recognition (SLR) system, feature extraction and recognition methods are needed. When talking about gesture recognition, one of the most complex challenges encountered is gesture segmentation, meaning knowing when a gesture starts and stops. Fortunately, many methods were developed to pass this challenge. (Suharjitom, 2018).

Recent research has focused on end-to-end deep learning models that directly map sign language videos to text or speech, eliminating the need for intermediate processing steps.

2.1. Machine Learning Model

A machine learning dataset is a compilation of data utilised for model training. It serves as a practical demonstration to instruct the machine learning algorithm on how to make accurate predictions. (Siddharth, 2012) The dataset was divided into training, validation, and test datasets. The training data represents the most critical subset, 60% of the total data. It assists in instructing the algorithm on what specific pattern or elements to identify within the data. After the model is trained, approximately 20% of the overall dataset is allocated for evaluating the model's parameters. The test dataset is used as input in the last stage of the training process and represents the previous 20% of the total data. This dataset has not been used before, so the model is not familiar with it, and it is used to determine the model's accuracy.

Deep learning techniques, particularly convolutional neural networks (CNNs) and recurrent neural networks (RNNs) have shown promising results in sign language recognition. CNNs are effective for extracting spatial features from sign language images, while RNNs are suitable for modelling temporal dependencies in sign sequences.

Google developed MediaPipe Hands, an accurate hand and finger tracking solution. It relies on Machine Learning to detect 21 landmarks. MediaPipe Hands employs a machine-learning pipeline composed of multiple interconnected models: palm detection and hand landmark models (Kaluri, 2017; Manakitsa, 2024).

In Fig. 1, an example of Media Pipe Hands example is presented.

Fig. 1. The 21 critical points of a palm detected by MediaPipe Hands (MediaPipe, online)

During model training, parametrised machine learning is supplied with curated data. This process aims to produce a model with optimised trainable parameters that minimise an objective function. The model's input parameters, also known as hyperparameters, are customisable so that the learning rate of the algorithm behind it will be tuned according to the dataset. Performance metrics, such as accuracy, measure how effectively the model has acquired the desired representation. They provide insights into the model's performance, indicating how well it has learned from the data. Enhanced model performance translates into various real-life advantages, such as increased revenue, lowered costs, or improved user experience. Model validation is required because every algorithm presents different performances based on datasets. If the accuracy score of the model is low, the hyperparameters can be changed until the desired accuracy level is reached. In other words, validating machine learning model outputs is essential to guarantee accuracy. A substantial amount of training data is employed during the training of a machine learning model. The primary purpose of conducting model validation is to enhance the quality and quantity of data by identifying areas for improvement.

3. Detailed Design and Implementation

3.1. Application architecture

The Client-Server architecture of the American sign language recognition application is presented in Fig. 2.

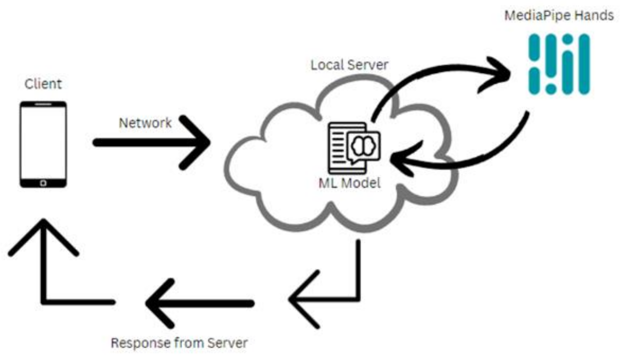

Fig. 2. Client–Server Architecture of the application

The client application is represented by the mobile application developed. The app user can upload an image or capture it using the device's camera, thus initiating a request to send the image to the server. The request is for type HTTP, specifically POST.

The server, running on localhost, listens for incoming requests and handles them by receiving the images sent by the client. The image is then sent to MediaPipe Hands API, which processes it and extracts the coordinates of the 21 critical points of a hand.

MediaPipe also draws on those coordinates' images to better visualise how the hand was perceived.

Next, the set of 3D coordinates, along with the image containing the landmarks, are sent back to the server where the trained machine learning model that can classify letters of the American sign language is hosted. Based on the coordinates received as input, the model categorises the letter.

Finally, the detected letter and the image with landmarks are sent back to the client as a response from the server. The user will see the reaction on the screen. This whole process can be repeated multiple times.

An important aspect to consider is the privacy of the application's users. Storing user-generated images can pose privacy and security risks, mainly if a virus attacks the server. Images may contain sensitive and personal information, and the user does not consent to storing or sharing their pictures. By not saving the photos, the risk of data breaches is minimised.

3.2. Sign language recognition model

The dataset can determine whether the model is good or not. The model's performance improves as the dataset size increases. An existing dataset from Keggle was chosen. It already contains about 87.000 images of the alphabet letters from the American sign language. In addition to this dataset, several hundred pictures collected in the laboratory environment were used.

The dataset was divided into two sets, with an 80% to 20% ratio. 80% of the data was used to train the model, while 20% was used to validate it.

When training the model, it was observed that MediaPipe Hands did not find all the images to contain hands. That's because although the dataset is numerous, the images are 200x200 pixels. Because of the poor quality of the photos, MediaPipe could not identify a hand in most of the pictures. To solve this issue and to avoid going through the entire dataset manually, a script was used to send the images to MediaPipe before the actual training. If a hand was detected, the respective image was saved in the "training_dataset" folder and labelled accordingly in the format 'letter' + 'image_number'. After this entire process, the training dataset consisted of only 25.000 images.

3.3. Model Training and Validation

Model training is the most crucial phase of developing a sign language recognition system employing machine learning techniques.

The model is formed from 6 layers: 1 input layer, two dropout layers, and three dense layers.

The training process involves using the fit function, where both the number of epochs and the validation data are specified, and the callbacks are used to perform specific tasks at specific stages during the training process. The total number of epochs is 10000, and the batch size is 4096 units. After the training phase, the model's performance can be evaluated based on the measured accuracy. From the logs received at the end of the training, we see that the model has an impressive accuracy score of 0.9019 and a loss score of 0.3267. In percentage, this means that the model's overall accuracy is 90%. In addition, the confusion matrix, see Fig. 3, was generated to conduct a more detailed analysis of the model's performance.

It was tested with new data using the mobile application to validate that the model was behaving as expected. In Fig. 4, two images are attached: the first represents the proof of the actual letter, while the second represents the model's response to the same image.

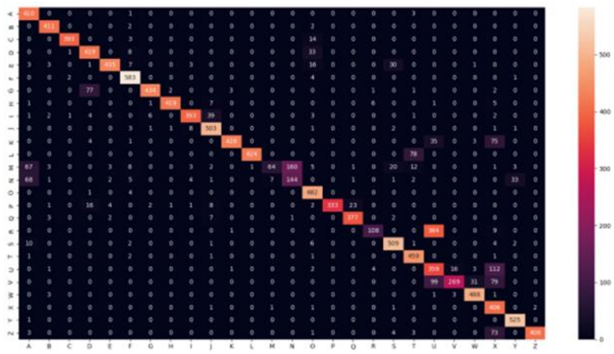

Fig. 3. Confusion matrix of the NN model

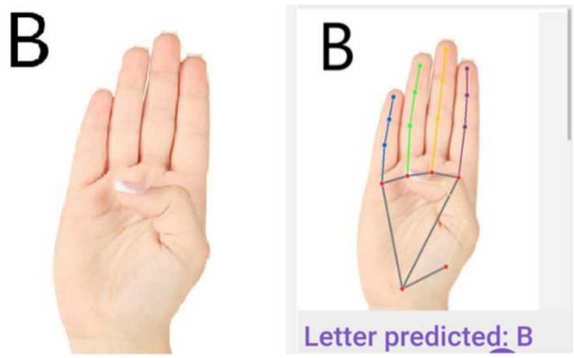

Fig. 4. Reference image vs Model prediction for B letter

After entering the application, the user is directed to the first page of the app, presented in Fig. 5. This page acts as the main page and has the functionality of a dashboard. A list of signs is displayed as the main content of the screen. Each sign is labelled with the scope of helping the user accommodate the letters and how they should be signed.

Fig. 5. The main screen of SignSense

Postman was used to test and validate that the server's services function accordingly. The testing method mainly focused on communication between the server and the client. In other words, the response to every request was tested.

We expect that every call responds with a 200-status code and that the server successfully receives the image sent as input from the request's header. The server's response can be observed even in the terminal in the form of logs.

During the implementation phase, client testing was done with an emulator provided by Android Studio. This helped mimic real-world scenarios without the need for an actual physical device. Also, because Flutter offers a hot reload functionality, it was beneficial from a time perspective because any change could be seen more easily on the emulator.

4. Conclusions

This paper has focused on developing a functional mobile application to recognise and classify American Sign Language (ASL) letters. By implementing the machine learning models based on CNN for signs classification, the accuracy was above 90%. The ML model training needs thousands of iterations for each sign. The application implemented in this paper makes a valuable contribution to advancing technology-based solutions to enhance communication and accessibility by harnessing the power of machine learning and mobile technology. It highlights that, through ongoing research and development in these fields, we can aspire for a future where technology-based solutions are easily accessible, user-friendly, and integrated into our everyday lives, enriching the experiences of all individuals.

Sign language in production processes is not only a communication tool, but also an important element building a safe, integrated and innovative workplace. The promotion and implementation of this language reflects the desire to create open, accessible and friendly development of technologies supporting sign language opens new possibilities for the integration and efficiency of production processes. Thanks to tools such as real-time translation applications or videoconferencing with interpreters, deaf and hard of hearing people can fully use their skills and competences, contributing to the success of the company.

This study highlights how technology-driven solutions can improve accessibility and communication for members of the deaf and hard-of-hearing community. Moreover, it represents a tool anyone can use to learn the basics of the American sign language: the alphabet. However, there is still room for improvement so that the application could fully satisfy the needs of the deaf community and their acquaintances.

The first improvement that comes to mind is extending the application's capabilities to perform real-time recognition of the American Sign Language from live video streams.

References

Adithya V. and Rajesh R.. 2020. *A deep convolutional neural network approach for static hand gesture recognition.* Procedia Comput. Sci. 171 (2020), 2353–2361.

ALTA (2024). *Deaf culture and sign language.* https://www.altalang.com/beyond-words/deaf-culture-and-sign-language/

Assael, Y. M., Shillingford, B., Whiteson, S., & de Freitas, N. (2016). *LipNet: Sentence-level Lipreading.* Computing Research Repository (CoRR), 1611.01599, 1-13.

http://arxiv.org/abs/1611.01599

Bhavsar, H. (2017). *Review of Classification Methods Used in Image-based Sign Language Recognition System.* International Journal on Recent and Innovation Trends in Computing and Communication, 5(5), 949–959.

Bhavsar, H., & Trivedi, J. (2017). *Review of feature extraction methods for image-based sign language recognition systems.* Indian Journal of Computer Science and Engineering (IJCSE), 8(3), 249-259.

Cao, Y., Chou, P. A., Zhang, Z., & Park, H. S. (2018*). Recent advancements in deep learning for sign language recognition: A review.* Computer Vision and Image Understanding, 171, 130-140.

Chai, X., Li, G., Lin, Y., Xu, Z., Tang, Y., Chen, X., & Zhou, M. (2013). *Sign language recognition and translation with Kinect.* In Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition (pp. 1-2). Shanghai, China.

Costa Filho, C. F. F., Souza, R. S. de, Santos, J. R. dos, Santos, B. L. dos, & Costa, M. G. F. (2017). *A fully automatic method for recognising hand configurations of Brazilian sign language.* Research on Biomedical Engineering, 33(1), 78–89.

Earthweb (2024). *Sign language users.* https://earthweb.com/sign-language-users/.

Elmezain, M., Al-Hamadi, A., Appenrodt, J., & Michaelis, B. (2008). *A hidden Markov model-based continuous gesture recognition system for hand motion trajectory.* In Pattern Recognition, 2008. ICPR 2008. 19th International Conference on (pp. 1–4). Tampa, FL, USA.

Escobedo, E., & Camara, G. (2016). *A new Approach for Dynamic Gesture Recognition using Skeleton Trajectory Representation and Histograms of Cumulative Magnitudes.* In SIBGRAPI Conference on Graphics, Patterns, and Images (pp. 209–216). São Paulo, Brazil.

Ghosh, D. K., & Ari, S. (2016). *On an algorithm for Vision-based hand gesture recognition*. Signal, Image and Video Processing, 10(4), 655–662.

Jayanthi Palraj, R K Ponsy, K Swetha, S A Subash. (2022). *Real-Time Static and Dynamic Sign Language Recognition using Deep Learning.* Journal of Scientific & Industrial Research 81(11):1186-1194. DOI: 10.56042/jsir.v81i11. 52657.

Kakde, M. U., Nakrani, M. G., & Rawate, A. M. (2016). *A Review Paper on Sign Language Recognition System For Deaf And Dumb People using Image Processing. International.* Journal of Engineering Research and Technology, 5(3), 590–592.

Kalsh, E. A., & Garewal, N. S. (2013). *Sign Language Recognition System*. International Journal of Computational Engineering Research, 3(6), 15-21.

Kaluri, R., & Pradeep, C. H. (2017). *An enhanced framework for sign gesture recognition using hidden Markov model and adaptive histogram technique.* International Journal of Intelligent Engineering & Systems,10(3), 11-19.

Manakitsa N, Maraslidis GS, Moysis L, Fragulis GF. (2024). *A Review of Machine Learning and Deep Learning for Object Detection, Semantic Segmentation, and Human Action Recognition in Machine and Robotic Vision.* Technologies.12(2):15. https://doi.org/10.3390/technologies12020015.

MediaPipe, "MediaPipe Documenation", [Online]. Available: https://developers.google.com/mediapipe/solutions/vision/hand_landmarker.

Naik BT, Hashmi MF, Bokde ND. (2022). A Comprehensive Review of Computer Vision in Sports: Open Issues, Future Trends and Research Directions. Applied Sciences. 2022; 12(9):4429. https://doi.org/10.3390/app12094429.

Nimratveer Kaur Bahia, Rajneesh Rani. (2023). *Multi-level Taxonomy Review for Sign Language Recognition: Emphasis on Indian Sign Language.* ACM Transactions on Asian and Low-Resource Language Information ProcessingVolume 22Issue 1Article No.: 23pp 1–39https://doi.org/10.1145/3530259.

Siddharth Rautaray, Anupam Agrawal, (2012). *A comprehensive review of vision-based hand gesture recognition.* Artificial Intelligence Review 43(1). DOI: 10.1007/s10462-012-9356-9.

Suharjitom F. W. G. P. K., Meita Chandra Ariesta. (2018). *A Survey of Hand Gesture Recognition Methods in Sign*. Pertanika Journal of Science and Technology, p. 19, 2018.

The Alliance for Access to Computing Careers (2024). *Deaf or Hard of Hearing.* https://www.washington.edu/accesscomputing/what-sign-language.