ISSN: 1692-7257 - Volume 2 – Number 46 - 2025

RCTA
Revista Colombiana de Tecnologías de Avanzada
UNIPAMPLONA

Digital Object Identifier: 10.24054/rcta.v2i46.4105

Sign language to text translation using Python with LSTM neural networks

Traducción de lenguaje de signos a texto mediante python con redes neuronales LSTM

Ing. Luis Duván Torrado Mora [1], **PhD. César Alberto Collazos** [2], **PhD. Dewar Rico Bautista** [1]

[1] ***Universidad Francisco de Paula Santander,*** *Ocaña, Norte de Santander, Colombia.*
[2] ***Universidad del Cauca,*** *Popayán, Cauca, Colombia.*

Correspondence: *dwricob@ufpso.edu.co*

Received: *april 24, 2025.* ***Accepted:*** *july 03, 2025.* ***Published:*** *july 25, 2025.*

How to cite: *L. D. Torrado Mora, C. Alberto Collazos, and D. Rico Bautista, "Sign language to text translation using Python with LSTM neural networks", RCTA, vol. 2, no. 46, pp. 150–159, Jul. 2025.*
Recovered from https://ojs.unipamplona.edu.co/index.php/rcta/article/view/4105

Abstract: there is a difficulty that deaf-mute people face in communicating effectively with those who do not know sign language. Despite the existence of methods such as writing and lip-reading, these have limitations and are not always effective. The proposed solution includes developing a real-time sign language recognition system using convolutional neural networks and the MediaPipe platform. It detects and classifies the positions of hand points to identify letters. Gestures made in front of the camera are translated into letters that are stored to form paragraphs in a text box. The type of research is quantitative and experimental. Ultimately, the importance of sign language recognition and teaching is highlighted, especially in countries such as Colombia, where it has received significant recognition.

Keywords: LSTM, mediapipe, gesture sign recognition, sign language.

Resumen: Existe la dificultad que enfrentan las personas sordomudas para comunicarse eficazmente con quienes no conocen el lenguaje de señas. A pesar de la existencia de métodos como la escritura y la lectura de labios, estos presentan limitaciones y no siempre son efectivos. La solución propuesta incluye el desarrollo de un sistema de reconocimiento de lenguaje de señas en tiempo real, utilizando las redes neuronales convolucionales y la plataforma MediaPipe. Detecta y clasifica las posiciones de los puntos de las manos para identificar letras. Los gestos hechos frente a la cámara se traducen en letras que se almacenan para formar párrafos en una caja de texto. El tipo de investigación es cuantitativa y experimental. Al final, se destaca la importancia del reconocimiento y la enseñanza del lenguaje de señas, especialmente en países como Colombia, donde ha recibido un reconocimiento significativo.

Palabras clave: LSTM, mediapipe, reconocimiento de signos, reconocimiento de gestos.

1. INTRODUCTION

Since ancient times, attempts have been made to communicate with deaf people with speech disabilities to integrate them into society, either by drawings, words, signs, or many other methods. People with the difficulty of not being able to speak or hear have many methods of communication which they have learned throughout their lives, for example, the mere fact of pointing at something is already a method of non-speech communication [1], [2]. The most formal model of communication for these people is sign language, which they learn through family members or, on the contrary, by watching other people speak sign language [3]. The communication of deaf people in a society with speech disabilities that do not know the language today is a bit confusing for both parties. Some deaf people with speech disabilities can use the means of writing, in the case of knowing how to write correctly, on the other hand, people with this condition often know how to read lips allowing them a little less complexity when communicating [4].

On many occasions the person with the condition of deaf-mutism implies the difficulty of not being able to hear what another person is transmitting, this leads to the end of reducing the probability of being able to understand another person, which causes a lot of difficulty to be able to write correctly. People with deaf mutism can write correctly but with a certain degree of difficulty in transmitting information [5]. One of the ways to overcome this problem is to use sign language [6], a real-time image recognition system capable of accurately identifying letters of the LIS alphabet provided by a user in human-computer interaction (HCI) framework using Python's Open-Source Computer Vision (OpenCV) library and two models based on convolutional neural networks, namely CNN and VGG19 [5]. Sign language just like speech or writing language is different in all countries, this is carried hand in hand with common language [4]. Allowing that in each country it is possible to train the software proposed here for the development of a problem such as communication with deaf people with speech disabilities [7].

A robust hand gesture recognition system based on high-resolution thermal images that are independent of light is proposed. A dataset of 14,400 thermal hand gestures separated into two color shades is constructed [4]. It is intended to use the user's hand as a region of interest (ROI) which will then be analyzed and processed to combat the effects of light and other anomalies that may arise, then use Artificial Neural Networks (CNNs) for learning [8]. The study of sign language in Colombia has been given significant importance since 1984 and was officially recognized in 1996 during the government of Ernesto Samper Pizano through law 324, in which the article reads as follows "The Colombian state recognizes the sign language as belonging to the deaf community of the country". The article shows the development and testing process of Colombian sign language translation software using a series of tools, such as artificial neural networks, machine learning, convolutional neural networks, and software for the use of these technologies [9], [10]. It uses an AI tool called Mediapipe for recognition of faces, objects, poses, hands, facial meshes, and object movement tracking [11].

Other projects are based on the translation of the sign by displaying a letter on the screen. A novel feature of the improved software is that it not only displays the translated letter on the screen but also outputs it by voice command [1], [6], [7], [12]. This article is an extension of the paper originally presented at the 10th Ibero-American Conference on Human-Computer Interaction JIHCI2024, June 4 to 7, 2024 Technological University of Pereira, Colombia [13].

The article is organized as follows. First, the introduction where the situation of the problem is presented. Second, is the methodology where the design of the research process to obtain and evaluate the proposed solution is presented, and the third part is the results, together with the respective discussion.

2. METHODOLOGY

The type of research will be quantitative since it refers to the approximation of reality understood as a goal and considers that the researcher must take distance from that reality to analyze it [14], [15]. The researcher considers that "the data must be discovered and analyzed objectively [16], [17], [18], [19] so that phenomena in a population are described, measured, explained, and predicted. An experimental methodology was implemented where tests are performed on a sign recognition software developed and an idea of a solution to understand deaf people with speech disabilities is proposed. The development is based on convolutional neural networks which allow the classification of images by layers and the recognition of the positions of the points of the hands, the software has the application of the open-source platform MediaPipe which has

3D object detection in real-time, so that detects 2D objects and estimates their poses through a machine learning model [20]. By recognizing these points, each gesture made in front of the camera will tell us in real-time which letter it refers to and this information will be saved so that the sequences of several gestures will form a paragraph in the text box that will be used to display the translation, (see Fig 1).

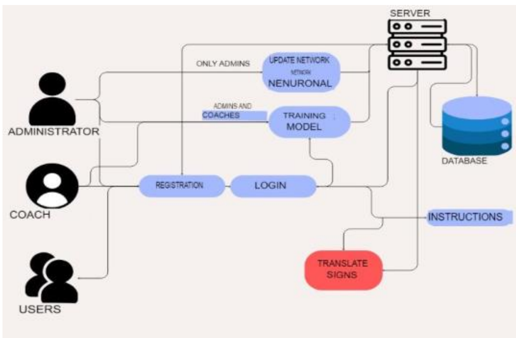

Fig. 1. *System Architecture: Client – Server.*

Code and structural designs are applied for the correct functioning of the neural network, where the user is thought of as a fundamental part of the development of existing and future technology, the software design is based on characteristics that define the user as a person with low knowledge of technology, should be oriented to all types of people and also be intuitive, these characteristics facilitate programming and understanding of how a deaf-mute person communicates [21], [22], [23], [24], [25], (see Fig 2).

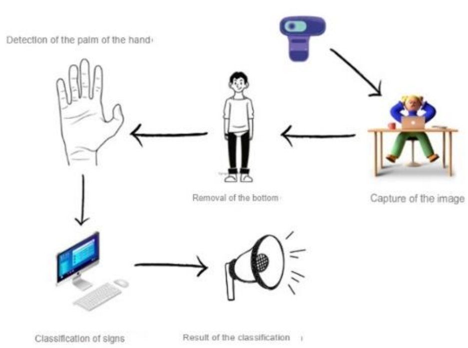

Fig 2. *Usability architecture: Client - Software.*

First, software training is performed for the recognition of each letter where three hundred images of the position of each letter are taken; second, the model and the weight of the letters are compiled and third, the prediction is performed. [26]. Another way of looking at the deployment of the project would be with Artificial Intelligence, neural networks (made up of convolutional), and mediapipe (sign language translation), [27]. In this way, the software can be implemented with the following steps:

- Development phase: Focuses on the use of technologies such as mediapipe for neural network training.
- Training phase: Where the three hundred images of the hand position concerning each letter are taken.
- Compilation phase: The process of compiling the images taken during training is performed.
- Validation phase: The model of the training phase is copied in which the two phases are compared to obtain an optimal result.
- Prediction phase: At the end, the letters can be translated concerning each hand position.
- Evaluation phase: Deaf people with speech disabilities take part in testing the software.

We understand that between two people to establish a communication there must be a sender and a receiver, the sender is in charge of sending a message and the receiver is in charge of receiving and understanding the message. Sign translator software can not only help deaf and mute people to establish communication but can also help other types of people, such as. Communication for deaf and deaf-mute people a sender with a hearing disability only wants to send a message to a deaf-mute person, and the receiver receives the message by reading the person's lips and responds using signs, the software takes those signs and translates them into text and voice command, achieving communication between sender and receiver, since even though the sender in this case is a deaf-mute person and the receiver is a person with a hearing disability, they can understand the message by reading the computer screen without having to listen to the voice command, (see Fig 3).

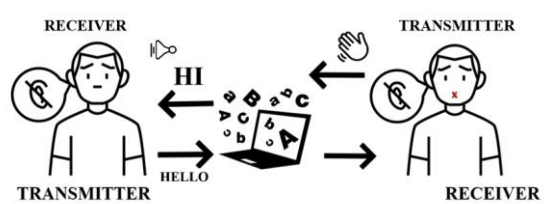

Fig 3. *Dialogue between a deaf-mute and a blind man.*

3. RESULTS

To understand how the software works, it is necessary to understand the branches of artificial intelligence and where the article is positioned, (see Fig. 4). The filtering of the images is done by artificial neural networks which classify the positioning of the fingers within the frame of each photo, thus performing the desired convolution [11]. The branches of artificial intelligence are divided into robotics, planning, natural language processing, computer vision, expert systems, automatic speech recognition, machine learning, and finally neural networks [28]. The latter are used to train the application by convolutions that are filters made to the images obtained in the training by a considerable number of samples. Having as a function the tracking of the hands through the platform, the free code called Mediapipe is used for the correct translation of sign language to text [29].

The handling and training of this software are conducted in three phases, which take some time for its correct translation [30]. The first is based on training the software by capturing three hundred images of the position of the hand for each letter of the alphabet; this amount is to achieve an accurate translation of each letter. The second, a compilation of the thirteen hundred images of each letter of the alphabet is performed [31]. In this compilation, a neural network is obtained, which will be used to link the camera with the image database, and the third is the real-time translation of the images of the hands captured by the camera. An update was made to the use of neural networks where a new network called LSTM is applied where better and more precise results are obtained.

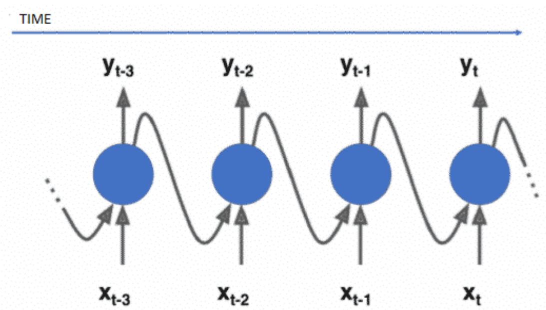

Fig 4. *LSTM neuronal network.* ***Source:*** *[11].*

3.1. LSTM

It is a type of neural network that is more efficient than others used in the development of this project, with better learning capacity and better response in less time. The LSTM model consists of layers that are sequential and consist of dropout layers that are responsible for preventing overfitting, and dense layers that are fully connected layers and have multiple outputs. Each LSTM cell consists of three main gates:

- **Input door**: It is what determines how much new information can enter or should be stored.
- **Output gate:** Controls how much information should leave the cell or be sent to the next cell.
- **Forget gate:** Decides how much information that has already passed should be forgotten by the cell.
- **Memory cell:** stores all the information over time

3.2. Data collection

This phase focuses on the collection of all the data that will be added to the project where it will be implemented sequentially based on the needs of the expected product. First, we start with the need to communicate through a computer to understand a deaf-mute person, from there we can record the behavior of these people, the means of communication they use, the environment in which they best defend themselves, or what the means of communication if a person cannot understand them, after recording this data we can begin with the beginning of the project in which we know that not all countries use the same sign language, with this information we can focus our project a little more allowing the use of common vocabulary words to speed up the management of other phases of development of the project where we require that our software not consume many image processing resources. Data collection is done through a mediapipe which detects the points of the hands and analyzes the position they are in to take the photos, (see Fig. 5).

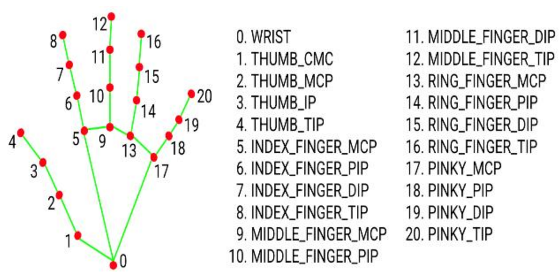

Fig 5. *Hand Landmarks Image.*

3.3. Development phase

In the development phase, a series of steps are proposed for the study of communication of deaf people with speech disabilities where certain questions are addressed.

- ¿How do deaf people with speech disabilities communicate?
- ¿What means of communication do deaf people with speech disabilities use?
- ¿Is sign language the same in all countries?
- ¿What technologies exist to enable communication between deaf and hard-of-hearing people and ordinary speech people?
- ¿Is there software that translates the sign and says it as a voice command?

3.4. Training phase

The data entered from the training of artificial neural networks helps to decipher each letter to be translated. Some letters are a bit confusing for the software, either because the fingers are positioned in the same way, or because some letters are performed by movement [32]. Figure 7 shows phase 1, which is the training phase of the software where three hundred images of the positioning of each letter with the hand were introduced. Dactylology should be executed with the person's most skilled hand, i.e., if right-handed, the right hand should be used, if left-handed, the left; when ambidextrous, only one of the hands should be selected, otherwise, it would confuse the receivers [26], (see Fig 6).

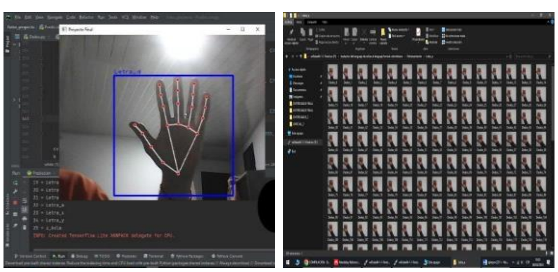

Fig. 6. *Software training a) Hand movement capture b) Different positions of each letter with the hand.*

The arm should be positioned comfortably, without any added effort. For the training of the network, three hundred images will be taken for each letter of the alphabet to generate less error percentage, it should be noted that sign language is not a universal language, each country has its own system. The background provided information about other projects already conducted and it is concluded that all of them are based on the neural network method [33]. It proposes a simple model that simulates the way the human brain processes information, which works by simultaneously combining many interconnected processing units that look like abstract versions of a neuron. To achieve assertive communication and a form of expression through the hands known as dactylological language seeking to capture each with the software letter to translate it achieving a way of inclusion [11]. It should be noted that the software not only serves to translate a letter of the alphabet, but can also be trained to achieve the capture of both hands to translate a word, a recommendation is that the software only captures static images, It is also intended to make improvements where not only the hands are captured, but also the face and torso of people to have more knowledge of the position, mood, and place in which it is for an assertive accuracy. This phase does not require training the model again, but only updates new words and improves the learning results within the neural network. This means that if we have a word and it is difficult for the software to detect it, new training is performed on the one that already exists and the quality of each word is improved.

3.5. Compilation phase

This phase does not have a graphical interface where you can see how the images are compiled, only the process of generating the neural network and the generation of files for use in the prediction phase is shown by the console [11]. The compile phase is based on taking the training images and forming a neural network in which all the letters of the alphabet are found. The main purpose is to give an overview of the best practices used in the development of software for sign recognition and to answer research questions that will contribute to the future development of the project, based on the training carried out in the first phase, different files are created which will help us in the next phase to be able to detect the samples introduced by the camera in real-time and also extract them through voice commands. These files are the ones that contain samples in matrices capable of translating a word or letter, (see Fig 7).

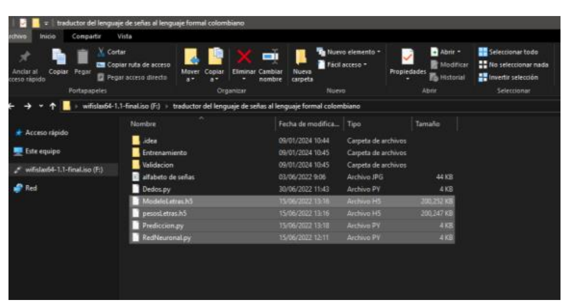

Fig 7. *Compilation phase.*

In this case, for example, we want to give recognition to the positioning of the hands and translate what has been previously trained in the software, the use of the software is required for educational purposes, means of communication in private and commercial places such as hospitals, health centers, schools, universities, banks, municipalities and places of government attention, in short, a software that can communicate with deaf-mute people. As the program uses matrices according to the number of words or letters that we introduce in the training phase, this matrix will work synchronized with the images captured by the camera, managing to show the matrix and the number of columns that are in it.

3.6. Validation phase

As in the training phase, images are taken from the camera for each position of each letter, where it is tried as much as possible to be the same as in the first phase to achieve better data accuracy of the points of the hands, since the software takes the photos of the training phase and performs a convolution with the photos of the validation phase, thus achieving to show the letter that is trying to translate, understanding the operation of the software is not complex, it is like having 2 photos and finding the difference in them, highlighting the result and showing it, the validation phase has the same photos of the training, the same amount of images for each letter and the same amount of letters, (see Fig. 8).

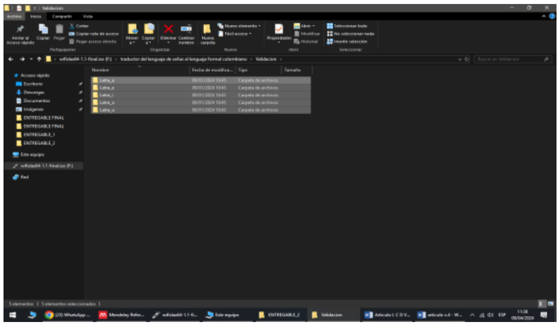

Fig. 8. *Validation phase.*

3.7. Prediction phase

The prediction process in which the software compares the images of the training with those of the camera, managing to take twenty-one points of the hands, obtaining a letter on the box in addition to obtaining through the speaker the letter that is being translated, (see Fig. 9).

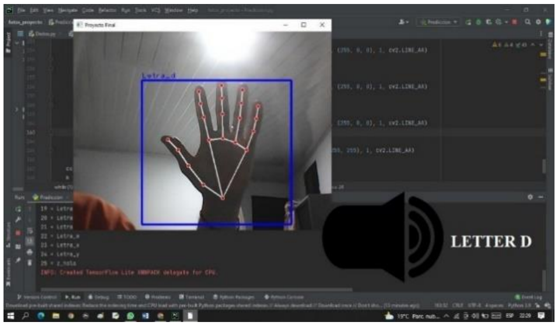

Fig. 9. *Compilation phase.*

Sign language translator software is possible because it does not require high resources or expensive machines, today you can use a computer and a webcam for the use of the application. The software will be able to translate the sign alphabet in real time and store the translated letters in a text box to assemble the desired word, and to avoid storing unwanted letters, the storage of the letters will be reflected through a 98% pass rate. The application will be built from mediapipe, an open-source tool with different functionalities, among which is the recognition of the twenty-one points of the hands. Mediapipe in coordination with Python uses machine learning which consists of several methods that are interconnected and work together. The first model is to detect the palm of the image that is considered as input. This model will return a hand-oriented bounding box bounding box of the captured palm image. The bounding box will be used to crop the hand image and then put inside the database [11]. For example, for vowel training, the software will create a matrix (A) of five columns where each column represents each vowel, this vowel represents the vowel that is being displayed on the camera which represents the letter U.

$$[0,0,0,0,1] = A \tag{1}$$

3.8. Evaluation phase

We plan to incorporate usability testing in which the relevant tests are performed, (see Fig. 10). In the evaluation phase of the previously trained model, the samples that are being taken in real-time are compared with the training samples. This is why it is important to perform training with precise data and good image quality to obtain optimal results. This phase is the final one, which allows the detection of faults in the model and, based on this, performs optimization of results. This, is to generate improvements to the software such as the recognition of hands in environments with variable light, different skin colors, different hand sizes, depth at which the hands are located, and other factors that could have a significant update of data from this [21], [22], [23], [24], [25], [34], [35].

Figure 10. *Project fair Universidad Francisco de Paula Santander Ocaña.*

3.9. Technical limitations

In the development of the project, different components were used to meet specific objectives. Several of these components present updates that are of great help for future advances on which we will rely in the future to implement new technologies and have greater benefit from the software. The use of these technologies helps to obtain the desired result, combining them with the translation of signs, such as the implementation of mediapipe, which allows us to detect the hands in certain frames of a sample from which the points of the fingers can be extracted. And the coordinates where they are located. At the beginning of the project, it was possible to translate only letters of the alphabet, having difficulties with the software since some letters were similar concerning the positioning of the hands and at that time somewhat slow technologies were being used for image processing, such as the use of CNN neural networks together with a computer with little capacity for sample collection and treatment. The current difficulties of the project are based on the versions of the libraries where only specific versions are used for correct operation since if new or old updates are used, erroneous results are obtained or the images registered or to be captured are not processed, for this reason, they are They use libraries with unique versions.

Tensorflow, Keras, and mediapipe among others are the applications or tools used, and which are shown below with their respective correct versions to work with the software.

Tensorflow==2.10.1, mediapipe==0.10.11, numpy==1.26.4, tables==3.9.2, OpenCV-python==4.9.0.80, pandas==2.2.1, protobuf==3.20.3, keras==2.10.0, h5py==3.10.0, flatbuffers==24.3.7, gTTS==2.5.1, pygame==2.5.2

4. CONCLUSIONS

To achieve effective detection and avoid letter confusion, the gestures of the fingerprint language must be perfectly defined. While current state-of-the-art approaches rely mainly on powerful desktop environments for inference, our method achieves real-time performance on a cell phone and even adapts to multiple hands. An application software was developed that emphasizes the learning of sign language, with digital image processing techniques in which the user must be able to coordinate the tilt, rotation, and movement to learn the symbols more easily.

It must be considered that the letters ñ and z are not static images for the program, they must be images with movement. It should be considered that the gestures of fingerprint language must be perfectly defined now as camera recognition, to avoid confusion between letters. The most used signs in sign language are from A to Z, considering that letters such as m, n, and r, have a certain degree of confusion in the program.

The development of the project is seen in the future as an aid platform for correct communication with deaf people with speech disabilities implemented in various places like shopping centers, schools, and banks. It will be able to translate sign language in both directions, from sign to text and from text to sign, without forgetting that the implementation of voice aids for blind people can also be done. The software could be improved by implementing automatic training and error correction with AI to achieve autonomous software in the implementation of new words or abbreviations of them as signs.

Different opinions about the development of the software have helped other developments of the project, such as, for example, people who are left without hands and have the condition of not being able to speak or hear, the implementation of eye tracking and an on-screen keyboard will be realized. The future use of innovative technologies in the software will help to conduct tests in various places to cover a large part of the population of Colombia where it would solve on a large scale the communication with deaf people with speech disabilities.

LSTM is better in response time compared to CNN neural networks, this is thanks to its ability to learn and respond in a short period. Despite advances in sign recognition accuracy, the software faces challenges in differentiating certain letters that have similar finger positions or involve movements. Therefore, improvements are planned that include the detection of facial expressions and torso positions to provide a more accurate and complete translation of sign language.

REFERENCES

[1] B. Villa, V. Valencia, and J. Berrio, "Digital image processing applied on static sign language recognition system/Diseño de un Sistema de Reconocimiento de Gestos No Móviles mediante el Procesamiento Digital de Imágenes," Prospectiva, vol. 16, no. 2, pp. 41–48, 2018, doi: 10.15665/rp.v16i2.1488.

[2] I. De Souza, P. Mattos, C. Pina, and D. Fortes, "ADHD: The impact when not diagnosed," J Bras Psiquiatr, vol. 57, no. 2, pp. 139–141, 2008, doi: 10.1590/s0047-20852008000200010.

[3] V. J. Schmalz, "Real-time Italian Sign Language Recognition with Deep Learning," in CEUR Workshop Proceedings, 2022, pp. 45–57.

[4] D. S. Breland, A. Dayal, A. Jha, P. K. Yalavarthy, O. J. Pandey, and L. R. Cenkeramaddi, "Robust Hand Gestures Recognition Using a Deep CNN and Thermal Images," IEEE Sens J, vol. 21, no. 23, pp. 26602–26614, Dec. 2021, doi: 10.1109/JSEN.2021.3119977.

[5] D. S. Breland, S. B. Skriubakken, A. Dayal, A. Jha, P. K. Yalavarthy, and L. R. Cenkeramaddi, "Deep Learning-Based Sign Language Digits Recognition from Thermal Images with Edge Computing System," IEEE Sens J, vol. 21, no. 9, pp. 10445–10453, May 2021, doi: 10.1109/JSEN.2021.3061608.

[6] I. A. Adeyanju, O. O. Bello, and M. A. Adegboye, "Machine learning methods for sign language recognition: A critical review and analysis," Nov. 01, 2021, Elsevier. doi: 10.1016/j.iswa.2021.200056.

[7] R. Rastgoo, K. Kiani, and S. Escalera, "Hand sign language recognition using multi-view hand skeleton," Expert Syst Appl, vol. 150, p. 113336, Jul. 2020, doi: 10.1016/j.eswa.2020.113336.

[8] H. Liu, Z. Zhang, J. Qian, W. Wang, and K. Kang, "Hand gesture recognition based on deep neural network and sEMG signal," in IEEE International Conference on Robotics and Biomimetics, ROBIO 2019, Institute of Electrical and Electronics Engineers Inc., Dec. 2019, pp. 3001–3006. doi: 10.1109/ROBIO49542.2019.8961445.

[9] J. A. Zea Gutiérrez, M. J. Suárez Barón, and J. S. González Sanabria, "Aprendizaje por refuerzo como soporte a la predicción de la precipitación mensual. Caso de estudio: Departamento de Boyacá - Colombia," TecnoLógicas, vol. 27, no. 60, p. e3017, Jun. 2024, doi: 10.22430/22565337.3017.

[10] L. Antonelli, M. Lezoche, and J. Delle Ville, "Knowledge Extraction from the Language Extended Lexicon Glossary Using Natural Language Processing," TecnoLógicas, vol. 27, no. 59, p. e2917, Apr. 2024, doi: 10.22430/22565337.2917.

[11] L. D. Torrado-Mora and D. Rico-Bautista, "Systematic Mapping: Translator Language from Sign Language to Colombian Formal Language," in Proceedings - 3rd International Conference on Information Systems and Software Technologies, ICI2ST 2022, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 44–48. doi: 10.1109/ICI2ST57350.2022.00014.

[12] S. Sharma and S. Singh, "Vision-based hand gesture recognition using deep learning for the interpretation of sign language," Expert Syst Appl, vol. 182, p. 115657, Nov. 2021, doi: 10.1016/j.eswa.2021.115657.

[13] L. D. Torrado-Mora, C. A. Collazos, and D. Rico-Bautista, "Sign Language to Colombian Formal Language Trans-lation Software," in Human-Computer Interaction. HCI-COLLAB 2024. Communications in Computer and Information Science, V. Agredo-Delgado, P. H. Ruiz, and C. A. Meneses, Eds., SPRINGER, 2024, pp. 1–14.

[14] R. Hernández Sampieri, C. Feránadez Collado, and M. D. P. Baptista Lucio, Metodología de la investigación. McGraw Hill España, 2014.

[15] L. A. Tovar, "Elaboración de tesis. Estructura y metodología," Trillas, p. 384, 2017.

[16] M. Gómez, Introducción a la metodología de la investigación científica, 1st ed. Argentina, 2015.

[17] G. Briones, Metodología de la investigación cuantitativa en las ciencias sociales, vol. 1,

no. 958-9329-09–8. 2002. doi: 10.1038/2191218a0.

[18] E. Del-Canto and A. Silva, “Metodología Cuantitativa: Abordaje desde la complementariedad en ciencias sociales,” Rev Cienc Soc, vol. 0, no. 141, p. 45, 2013, doi: 10.15517/rcs.v0i141.12479.

[19] A. Radmehr, M. Asgari, and M. T. Masouleh, “Experimental Study on the Imitation of the Human Head-and-Eye Pose Using the 3-DOF Agile Eye Parallel Robot with ROS and Mediapipe Framework,” in 2021 9th RSI International Conference on Robotics and Mechatronics (ICRoM), IEEE, Nov. 2021, pp. 472–478. doi: 10.1109/ICRoM54204.2021.9663445.

[20] S. Adhikary, A. K. Talukdar, and K. Kumar Sarma, “A Vision-based System for Recognition of Words used in Indian Sign Language Using MediaPipe,” in Proceedings of the IEEE International Conference Image Information Processing, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 390–394. doi: 10.1109/ICIIP53038.2021.9702551.

[21] T. Granollers i Saltiveri, “MPIu+a. Una metodología que integra la Ingeniería del Software, la Interacción Persona-Ordenador y la Accesibilidad en el contexto de equipos de desarrollo multidisciplinares,” 2004.

[22] J. Lorés and T. Granollers, “Ingeniería de la Usabilidad y de la Accesibilidad aplicada al diseño y desarrollo de sitios web,” no. May, pp. 3–7, 2004.

[23] T. Granollers, “Usability Evaluation with Heuristics . New Proposal from Integrating Two Trusted Sources 2 Combining Common Heuristic Sets,” pp. 1–16, 2018.

[24] T. Granollers, “Diseño Centrado en el Usuario (DCU). El modelo MPlu+a,” p. 71, 2013.

[25] S. Inform and L. Lleida, “MPIu + a . Una metodología que integra la ingeniería del software, la interacción persona-ordenador y la accesibilidad en el contexto de equipos de desarrollo multidisciplinares,” 2004.

[26] S. Ikram and N. Dhanda, “American Sign Language Recognition using Convolutional Neural Network,” in 2021 IEEE 4th International Conference on Computing, Power and Communication Technologies, GUCON 2021, Institute of Electrical and Electronics Engineers Inc., Sep. 2021. doi: 10.1109/GUCON50781.2021.9573782.

[27] A. Radmehr, M. Asgari, and M. T. Masouleh, “Experimental Study on the Imitation of the Human Head-And-Eye Pose Using the 3-DOF Agile Eye Parallel Robot with ROS and Mediapipe Framework,” in 9th RSI International Conference on Robotics and Mechatronics, ICRoM 2021, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 472–478. doi: 10.1109/ICRoM54204.2021.9663445.

[28] V. Chunduru, M. Roy, N. S. Dasari Romit, and R. G. Chittawadigi, “Hand Tracking in 3D Space using MediaPipe and PnP Method for Intuitive Control of Virtual Globe,” in IEEE Region 10 Humanitarian Technology Conference, R10-HTC, Institute of Electrical and Electronics Engineers Inc., 2021. doi: 10.1109/R10-HTC53172.2021.9641587.

[29] N. F. Thejowahyono, M. V. Setiawan, S. B. Handoyo, and A. H. Rangkuti, “Hand Gesture Recognition as Signal for Help using Deep Neural Network,” International Journal of Emerging Technology and Advanced Engineering, vol. 12, no. 2, pp. 37–47, Feb. 2022, doi: 10.46338/ijetae0222_05.

[30] S. Njazi and S. Ng, “Veritas: A Sign Language-To-Text Translator Using Machine Learning and Computer Vision,” in ACM International Conference Proceeding Series, Association for Computing Machinery, Nov. 2021, pp. 55–60. doi: 10.1145/3507623.3507633.

[31] M. L. Wehmeyer et al., “The intellectual disability construct and its relation to human functioning,” Intellect Dev Disabil, vol. 46, no. 4, pp. 311–318, Jan. 2008, doi: 10.1352/1934-9556(2008)46[311:TIDCAI]2.0.CO;2.

[32] S. Srivastava, A. Gangwar, R. Mishra, and S. Singh, “Sign Language Recognition System Using TensorFlow Object Detection API,” in Communications in Computer and Information Science, Springer Science and Business Media Deutschland GmbH, Jan. 2022, pp. 634–646. doi: 10.1007/978-3-030-96040-7_48.

[33] S. Gulati, A. K. Rastogi, M. Virmani, R. Jana, R. Pradhan, and C. Gupta, “Paint/Writing Application through WebCam using MediaPipe and OpenCV,” in Proceedings of 2nd International Conference on Innovative Practices in Technology and Management, ICIPTM 2022, Institute of Electrical and Electronics

Engineers Inc., 2022, pp. 287–291. doi: 10.1109/ICIPTM54933.2022.9753939.

[34] T. Granollers, "Usability evaluation with heuristics. New proposal from integrating two trusted sources," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Springer Verlag, 2018, pp. 396–405. doi: 10.1007/978-3-319-91797-9_28.

[35] J. Lorés Vidal and T. Granollers, "La Ingeniería de la usabilidad y de la accesibilidad aplicada al diseño y desarrollo de sitios web," 2004.