IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 4 August 2023, accepted 2 September 2023, date of publication 6 September 2023, date of current version 14 September 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3312636

APPLIED RESEARCH

Sign4all: A Low-Cost Application for Deaf People Communication

FRANCISCO MORILLAS-ESPEJO[1], (Member, IEEE), AND ESTER MARTINEZ-MARTIN[1,2], (Senior Member, IEEE)

[1]RoViT Laboratory, Department of Computer Science and Artificial Intelligence, University of Alicante, 03690 Alicante, Spain
[2]University Institute for Computer Research (IUII), University of Alicante, 03690 Alicante, Spain

Corresponding author: Ester Martinez-Martin (ester@ua.es)

This work was supported by the Ph.D. Grant from the University of Alicante, Spain, under Grant UAFPU21-78.

ABSTRACT One of the main barriers for deaf people is communication due to the lack of understanding between them and the hearing society. This fact can considerably affect their daily life by leading to their social exclusion. On the way to an inclusive society, this paper presents a low-cost application to assist people to communicate by means of the Sign Language alphabet. For that, this application includes two functionalities: 1) a Sign Language recognizer, which is in charge of typing the letters being signed by deaf people; and 2) a virtual avatar signing what is written letter by letter. So, a comparative analysis of different techniques resulted in an accuracy of 79.96% when using the Convolutional Neural Network ResNet50 for sign recognition from an RGB image.

INDEX TERMS Virtual avatar, sign language, computer vision, teaching system.

I. INTRODUCTION

According to the Spanish National Institute of Statistics [1], more than 13% of the Spanish population suffers some sort of hearing impairment.

This condition creates a communication barrier between the hearing population and the deaf one, which often limits their capabilities to participate in an egalitarian society. In fact, this lack of communication could lead to a social, cultural and even labor marginalization.

One communication barrier is the access to information and media since deaf people have many difficulties in learning to read and write. Actually, 80% of deaf population does not reach suitable language proficiency [2]. This percentage rises to 92% of deaf population in the case of the Spanish society, leading to an economical inactivity percentage greater than 50% [3].

Other examples can be found in [4].For instance, Erica lost her child during pregnancy due to a misunderstood with the doctors; while Tilgen had to leave his city because Sign Language was not taught in schools.

The associate editor coordinating the review of this manuscript and approving it for publication was Ramakrishnan Srinivasan.

To overcome these situations, different technologies have been developed to enable people with hearing loss to hear sounds and, consequently, to learn both sign language and spoken language, as discussed by Napoli et al. [5]. Despite these technologies prevent marginalization in both hearing and non-hearing environments, their use is not always possible or fully functional for all deaf population. Instead, they use Sign Language as their primary method of communication.

As a consequence, the development of Sign Language recognition systems is acquiring great interest in the scientific community.

Although a generic Sign Language recognition model cannot be developed since each country has its own Sign Language (e.g. *Lengua de Signos Española (LSE)* in Spain, *Lingua dei Segni Italiana (LIS)* in Italy, *American Sign Language (ASL)* in the USA). This makes it necessary to learn the signs and grammar of each language. Furthermore, the lack of standardization within the country, together with the linguistic variations inherent to any language, gives rise to lexical differences between regions, as stated by Báez-Montero and Fernández-Soneira [6].

In everyday situations, misunderstandings may arise between hearing individuals and Sign Language users, such

as during medical appointments or at restaurants as described in [4]. Although interpreters are typically employed during these and other events (e.g. social events, meetings, or news broadcasts), there are instances where their services are not available. This could be due to different reasons, such as a short communication that may not justify the expense of hiring an interpreter, long waiting periods for an interpreter, or the need for an interpreter to be present at short notice without prior arrangement. To address this issue, this paper proposes a low-cost application to assist deaf people in communicating with hearing people. So, the system is designed to aid deaf people in everyday situations where the presence of an interpreter is not possible.

For that, two functionalities have been developed: a Spanish Sign Language (LSE) recognizer, capable of identifying what is signed by a deaf person and writing it into Spanish text; and a virtual avatar to sign what is written by a hearing person.

The recognizer functionality was trained with data relative to the Spanish Sign Language alphabet, which is used to sign proper nouns, streets, trademarks, or words that do not have a proper sign or whose sign is unknown.

In addition, there are different words whose handshape derives directly from the manual alphabet [6]. From a research perspective, the LSE alphabet is composed by a combination of static signs and those that require movement to be performed. This leads to the study of different deep learning architectures to understand both spatial and temporal dimensions of the data.

The virtual avatar was designed to sign the different letters composing the Spanish Sign Language alphabet. Thus, a database of the different visual letters has been developed.

This paper is structured as follows: Section II contains the previous works found in the literature; Section III describes the recognition engine used for *translating* between LSE and written Spanish; Section IV explains the creation process of the virtual avatar used to translate from written Spanish to LSE; Section V shows the results obtained with the complete system; and Section VI presents the conclusions and future work.

II. STATE OF THE ART

Sign Language Recognition has become a popular research field in the last decade. In this sense, several approaches have been studied.

Since hand tracking is the primary source of information in Sign Language, it is critical to solve this. Two different approaches have been identified to address this issue. On the one hand, there are approaches based on the use of gloves with specific sensors to track hand movements [7], [8], [9], [10], [11]. These methods provide parameters such as hand and finger position, orientation and/or movement with great accuracy. However, they require the use of specific hardware, what constrains the free movement of the user and the system's applicability. In addition, this kind of hardware could be very costly. So, these systems are unusable in real situations. Thus, new alternatives such as those based on computer vision must be analyzed.

Some classic computer vision alternatives used skin segmentation with Kalman filters[1] to distinguish between arms and hands [13], colored gloves on each finger [14] or a combination of cameras [15] to avoid occlusions. The main problem of these solutions is the system customization, what requires an adjustment for each signing person in terms of skin, glove color or camera position.

Furthermore, some alternatives based on depth sensors can be found in the literature. For instance, Kumar et al. [16] proposed a Sign Language Recognizer for 30 signs of the Indian Sign Language (ISL). For that, they used a Kinect sensor to extract the 3D skeleton pose of the user, and, from this, they obtained hand features. Using the extracted features as input, a Hidden Markov Model (HMM)[2] was used for recognition, achieving a top accuracy of 83.77% on their own dataset. In addition, thanks to the 3D values, they made the system robust to rotation and translation.

More recently, different deep learning techniques are proposed for Sign Language Recognition [18], [19], [20]. So, Triwijoyo et al. [21] presented a recognition system for the American Sign Language (ASL) alphabet based on Convolutional Neural Networks (CNNs).[3] For that, they used a dataset [23] consisting of 3, 000 hand images for each of the 26 signs that compose the ASL alphabet, and a 7-layer CNN model to recognize them. Their model achieved a theoretical 99% of accuracy, but in real-world scenarios the system is highly dependent on the light conditions, camera specifications and camera position, what considerably reduces the real accuracy.

Regarding the Spanish Sign Language, Rodríguez-Moreno et al. [24] proposed a system able to recognize 5 words from the Spanish Sign Language: *bien* (well), *contento* (happy), *mujer* (woman), *hombre* (man) and *oyente* (hearing person). They proposed a deep learning model to recognize the first and last configuration of a given sign. By using this strategy, they were able to convert a dynamic sign into two static ones (the first and last pose of each sign). Thus, no temporal relationship of the signs was required. Instead, only the spatial dimension was evaluated.

Given that Sign Languages are different one from another, their respective alphabets are also different. So, some of the studied languages in recognition research (i.e. Indian Sign Language (ISL), Arabic Sign Language (ArSL)) have their alphabets composed only by static signs, that is, no movement is necessary to perform them. However, others Sign

[1] A Kalman filter is an algorithm used in control and signal processing to predict a system's state based on noisy measurements [12].

[2] Hidden Markov Models are statistical models used to represent systems that probabilistically switch between unobservable states, emitting measurable observations associated with those states. They are used for processing and analyzing data sequences [17].

[3] Convolutional Neural Networks are deep learning models mimicking how a human brain processes visual information. They use filters to extract features from images, enabling the recognition of complex patterns [22].

TABLE 1. Summary of approaches in literature for Sign Language recognition.

Sign Language	Authors	Approach	Accuracy	Num. signs
Indian Sign Language	Kumar *et al.* [16]	HMM	83.77% (val)	30
Arabic Sign Language	Kamruzzaman [19]	CNN	90% (val)	31
American Sign Language	Triwijoyo *et al.* [21]	CNN	99% (test)	26
Spanish Sign Language	Rodríguez-Moreno *et al.* [24]	HMM	80% (test)	5
Italian Sign Language	Pigou *et al.* [20]	CNN	91.7% (val)	20

Languages (i.e. American Sign Language (ASL), Chinese Sign Language (CSL), Italian Sign Language (LIS)) have only two dynamic signs whose spatial position is sufficiently different to distinguish them without temporal information. This is not the case with the Spanish Sign Language alphabet, since it contains dynamic signs with a similar spatial position to the static ones, and there are more than two dynamic signs. Table 1 briefly summarizes the current state of the art in terms of sign recognition by indicating the studied Sign Language, the proposed approach, the number of signs to be recognized, and the achieved accuracy on the validation (val) or test subset.

Regarding the virtual avatar, some examples can be found in the literature. For instance, Hu et al. [25] presented a virtual avatar for the Chinese Sign Language in which they combined hand, body and leap movement. Their virtual avatar for Sign Language interpretation was presented as an standalone application and as an add-on for TV broadcasting. They used a combination of algorithms to implement an interpolation between the end and start points of two consecutive frames. One of the main drawbacks is the use of motion capture devices and professional interpreters to collect the signs that make up their dataset, making it difficult to add new data.

Another example is that proposed by Bhatti et al. [26], where they presented a virtual avatar with predefined gestures centered in three Sign Language alphabets: Sindhi (52 letters), Urdu (39 letters), and English (26 letters). They proposed the system as a learning platform or as a quick response system, where the user chooses the word to be signed and the virtual avatar signs it. Their system has some drawbacks, such as the lack of a module for transcribing Sign Language into written language or the avatar's use of a blueish-colored skin that does not resemble a natural or human-like appearance.

Other authors presented different approaches where they used a written input and translated it into the corresponding sign, such as [27] for the German Sign Language, [28] for the Italian Sign Language.

Regarding the Spanish Sign Language, two approaches can be found. Vera et al. [29] developed a virtual avatar to help in a classroom. As a consequence, it has a specific vocabulary related to that class and situation. Another approach was presented by López-Ludeña et al. [30]. It is mainly focused on the development of a speech to text synthesizer, but their avatar performs each sign by introducing a hand shape and orientation instead of a word. Both of them are closed and discontinued projects since 2015. This literature review is summarized in Table 2 by indicating authors, Sign Language and the number of signs.

TABLE 2. Summary of Virtual avatars in the literature.

Sign Language	Authors	Num. of signs
Chinese Sign Language	Hu *et al.* [25]	2,000 signs
Sindhi Sign Language	Bhatti *et al.* [26]	52 letters
Urdu Sign Language	Bhatti *et al.* [26]	39 letters
English Sign Language	Bhatti *et al.* [26]	26 letters
German Sign Language	Simax project [27]	Not specified
Italian Sign Language	Lombardo *et al.* [28]	Not specified
Spanish Sign Language	Vera *et al.* [29]	2,000 signs
Spanish Sign Language	López-Ludeña *et al.* [30]	653 signs

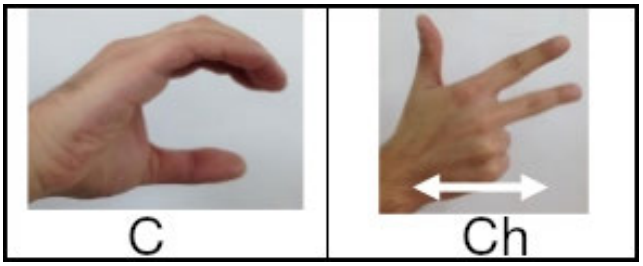

FIGURE 1. Left: *static* sign; Right: *in-motion* sign.

III. LSE TO TEXT ENGINE

As mentioned above, the goal of this research is to create an application for deaf communication. For this, it is necessary to develop two functionalities: one in charge of the recognition of the letters to convert them into written Spanish, and another aimed to the use of a virtual avatar to sign the written letters into Spanish Sign Language.

As previously stated, the system must be able to recognize the letters that composes the LSE alphabet. So, an improved version of our recognition system [31] was used. A notable modification introduced is the change from OpenPose [32] to MediaPipe [33] since Mediapipe provides a greater precision in the extraction of the skeleton as stated by Chung et al. [34].

In addition, a new dataset was recorded to increase the previous one from 8, 000 images to nearly 28, 000 (more than 3 times). To increase the number of images, six people (five men and one woman) additional to the original five (three men and two women) were recorded.

Although the dataset is out of the scope of this paper, it is worth mentioning that in the case of the alphabet, there are two types of signs: *static* and *in-motion*. The main difference between them is if the user signs in a specific position (*static*) or if it is necessary to perform some trajectory to complete the sign (*in-motion*), as depicted in Fig. 1.

To recognize the different visual letters, two deep learning techniques were studied: Convolutional Neural Networks (CNN) [35] and Recurrent Neural Networks (RNN).[4] Because the alphabet is composed of *static* and *in-motion* letters, the importance of the spatial dimension (CNNs) over the temporal one (RNNs) was studied.

[4]Recurrent Neural Networks are machine learning models used for processing sequential data. They have feedback connections to remember past information so that they are used in problems where data order matters [36].

According to these techniques, a prior study was realized in [31] where some of them were compared. This study was updated and completed in [37], concluding that the best three architectures were:

- ResNet50 [38]: This is one of the most common architectures in image classification tasks. It introduced the concept of residual blocks used to avoid the vanishing gradient in deep networks. These blocks are composed by identity connections that skip one or more layers to obtain identity maps without adding extra parameters nor computational complexity. In this case, a 50-layer architecture was used.
- LSE-CNN: This is one of our proposed architectures based on CNNs. As illustrated in Fig. 2, it is a 12-layer architecture that combines five convolution operations followed by a maxpooling layer each one. The convolution filters are increased in each layer from 16 to 256 while the kernel size remains the same, at 3×3. Note that the max pooling layers use a pool size of 2×2. The last layer is a fully connected one with the number of alphabet letters outputting the corresponding letter. Note that the optimizer Adam [39] with a learning rate of 0.001 and categorical crossentropy[5] as loss function[6] was used.
- LSE-RNN: LSTM (*Long Short-Term Memory*) [40] units are the most well-known RNN architectures. They are composed of a cell state and three gates that allows them to remember information over time. In particular, an extension of these units was used, Bidirectional LSTMs [41]. The main difference is that Bidirectional LSTMs access to past, present and future information, adding additional context to the network and improving its training. Thus, the proposed architecture is composed of two of these layers with 32 units followed by a fully connected layer to perform the classification task. It uses categorical crossentropy as loss function and Adam optimizer with a learning rate of 0.001.

These architectures were trained by using the hyperparameters summarized in Table 3.

TABLE 3. Hyperparameters used for LSE-CNN and LSE-RNN.

Architecture	Epochs	Optimizer	Learning rate	Loss function
LSE-CNN	100	Adam	0.001	Categorical crossentropy
LSE-RNN	100	Adam	0.001	Categorical crossentropy

Although the temporal dimension (RNNs) was considered to see if it would improve the results, especially in the *in-motion* letters, the experimental results showed that this was not the case (see Table 4). These results highlighted the importance of the spatial dimension (CNNs) over the temporal one (RNNs) in the Spanish Sign Language alphabet.

This happens because the spatial location of the signer's hand is quite different between letters, even if they have movement. So, with the letter location and shape, the deep learning models are capable to distinguish between them without the temporal context. Also, there are more *static* than *in-motion* signs, 18 versus 12. Furthermore, it should be mentioned that the most confusing letters in the network are the pairs *i-y*, *l-ll*, *n-ñ*, *r-rr* and *u-v*, whose main difference lies in the hand movement (see Figure 3).

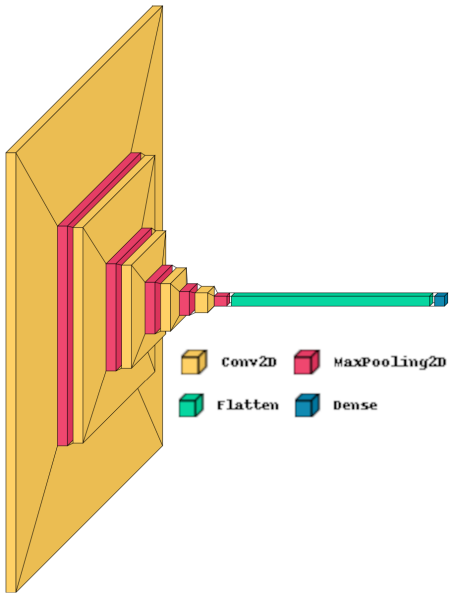

FIGURE 2. Proposed CNN architecture.

TABLE 4. Results in Spanish Sign Language alphabet recognition.

Model	Acc (val)	Acc (test)	Recall	Precision	F1 score
ResNet50	76.74%	78.81%	78.25%	80.36%	79.29%
LSE-CNN	57.62%	60.91%	55.32%	52.73%	54.00%
LSE-RNN	46.71%	51.64%	50.16%	48.59%	49.36%

IV. VIRTUAL AVATAR

Since the second functionality is the transcription from written Spanish to Spanish Sign Language, a virtual avatar was used for this purpose. Thus, an avatar was created modelling the upper part of a person, as illustrated in Fig. 4.

This avatar was designed to emulate a human interpreter as closely as possible. A realistic humanoid was created, but without falling into the uncanny valley phenomenon.[7] In addition, it wears a black t-shirt, since this is the color commonly used by Sign Language interpreters. It should be

[5]Categorical crossentropy is used for multi-class classification problems. It measures the discrepancy between model predictions and actual labels.

[6]Loss function is a measure to evaluate how well a model classifies or predicts values by comparing its predictions with the original labels. The goal is to minimize the loss function during the training to improve the model's performance.

[7]This phenomenon is a hypothesized relation between an object's degree of resemblance to a human being and the emotional response to the object. The concept suggests that humanoid objects that imperfectly resemble actual human beings provoke uncanny or strangely familiar feelings of uneasiness and revulsion in observers [42].

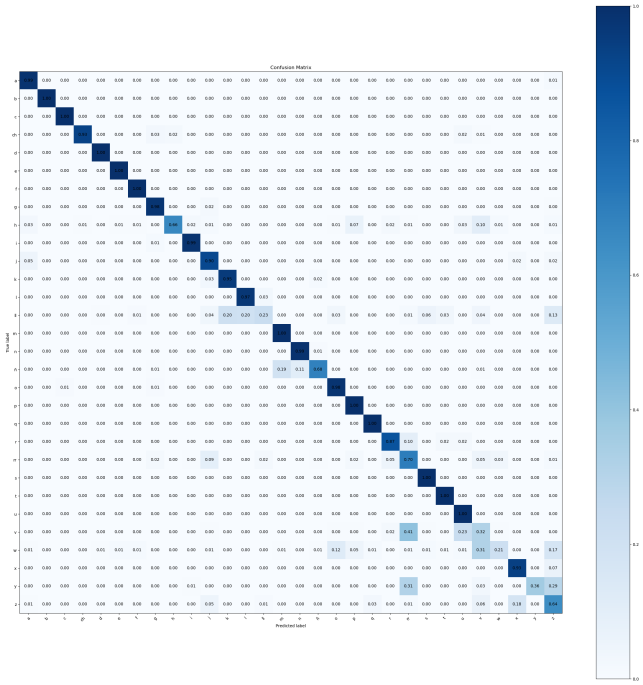

FIGURE 3. Confusion matrix for alphabet recognition in ResNet50 model.

FIGURE 4. Virtual avatar.

noted that the parts involved in the signing process, i.e. the hand and arms, were oversized to allow clearer movement in the animation.

A. AVATAR CREATION

During the creation process two different applications were used: Blender [43] and MakeHuman [44]. Blender is a 3D creation suite used to create and animate the avatar, while MakeHuman is used to build humanoid models. Both are free and open source applications, which allows them to be used for any purpose.

The first step in creating the avatar was to model its shape using MakeHuman. MakeHuman provides a humanoid base model where the different parts of the body can be modified (see Fig. 5), for example the shape of the arms and hands. As mentioned above, these variables were changed by elongating the arms and scaling up the hands and fingers for better visualization.

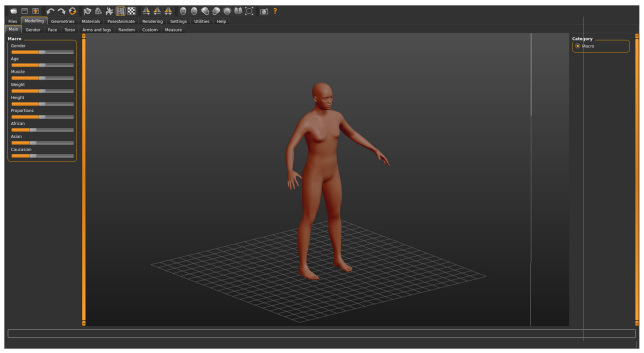

FIGURE 5. MakeHuman base model.

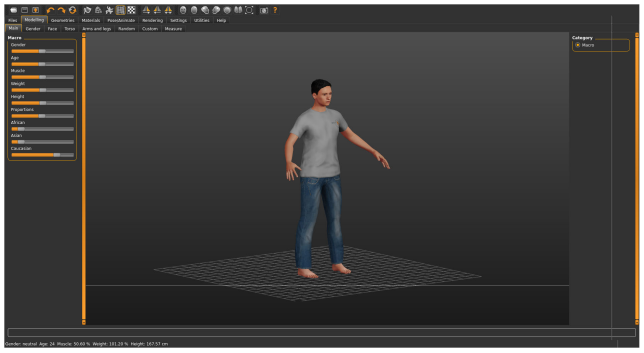

FIGURE 6. Complete person model.

After the baseline model was created, some clothes and hair were added to make it look like more real (Fig. 6). These are textures added to the model that combine shadow and light effects to 2D images, creating a 3D sensation.

Once the clothes were added, the 3D model was complete. It was then imported into the Blender application to start with the animation process.

B. AVATAR ANIMATION

With the 3D model of the avatar defined, it is necessary to create a skeleton for the body. This skeleton consists of solid parts (bones) connected between them by joints. They are similar to a human skeleton, i.e. to move the arm it is necessary to concatenate rotations of the joints that link each bone that composes it. Note that, in order to move a body part, the bones must be connected to their corresponding ''muscle'' in the 3D model; if they are not connected, the bones will move, but the model will remain in its default position.

The skeleton was created using MakeHuman, which allows the selection of several skeletons depending on the body parts to be animated: a skeleton optimized for motion capture with 31 bones, another for videogame animation with 53 bones, and finally the model for complex animations with 163 bones in its complete version and 137 in the version without feet. Since full control of the finger bones and no toes were required, the latter was used as shown in Fig. 7.

Once the 3D model and the skeleton were built and connected, they were transferred from MakeHuman to Blender,

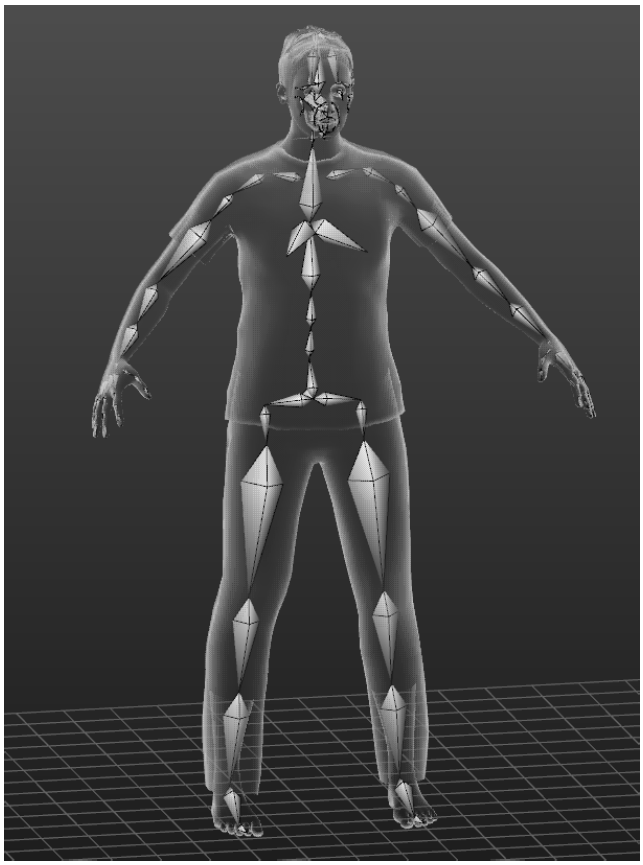

FIGURE 7. Skeleton made with MakeHuman.

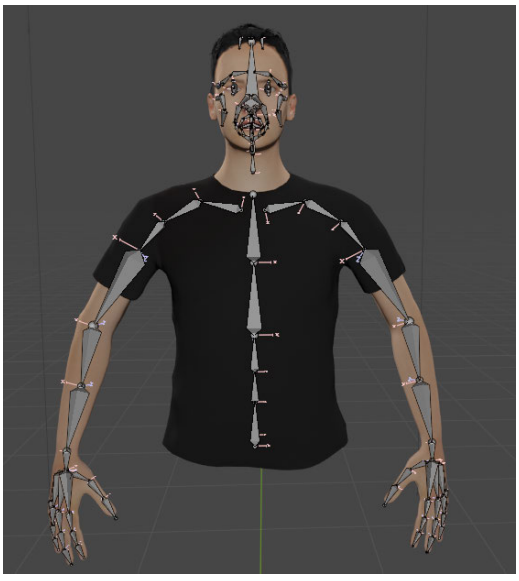

FIGURE 8. Upper version of the avatar.

which handled the animation process. Furthermore, the lower part of the body and the corresponding bones were deleted, resulting in the avatar depicted in Fig. 8.

To animate the character, each individual bone is rotated along the X, Y and/or Z axis as shown in Fig. 9. The

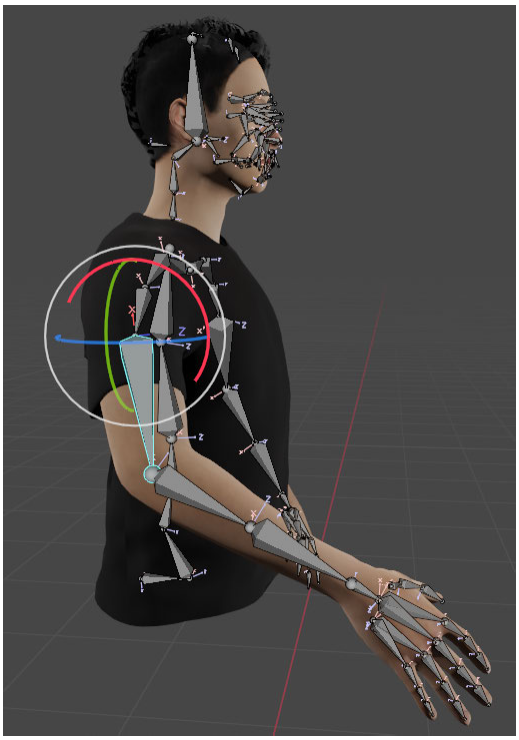

FIGURE 9. Axis movement: red for X, green Y, and blue for Z.

equations 1, 2, 3 show the mathematical functions of the rotations around each of the axis used by Blender; where, θ represents the rotation angle in radians.

$$rotX(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix} \quad (1)$$

$$rotY(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \quad (2)$$

$$rotZ(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

To reach a particular position, different bones must be rotated. For example, to close the left hand, first rotate the distal phalanges of each finger to face down, then do the same with the middle phalanges, and finally repeat the process with the proximal phalanges.

The movement of the joints are not limited, they can move freely and reach some impossible positions for humans (Fig. 10). If these movements are recorded at some point, the trajectory of the whole character will look artificial, so to get a more human-like movement, all the joints should move as humans do.

To create the animation, the arms and hands must be at characteristic points on the trajectory. These points are stored as key frames in the animation, and by interpolating between them the complete trajectory is obtained. To generate these points, small rotations were made in the various axes over

FIGURE 10. Examples of invalid movements.

successive joints; when the desired position was reached, the points were saved as key frames.

The distance between these key frames affects the speed of the animation. Blender uses interpolation to determine the position, rotation and scale of an object between key frames, which means that the distance between key frames determines the length of the line connecting those two points. A small distance between key frames results in a faster movement, while a larger distance results in a slower movement.

For all the signs, the same neutral and middle positions (Fig. 11) were saved at the beginning and ending of each sign. This is done to ensure that each sign has the same positions. The neutral (or rest) position is used as the start and end point, while the middle position is used as the transition between signs. In addition, the necessary points, which vary per sign, were stored as shown in Fig. 12 for the letter *x*.

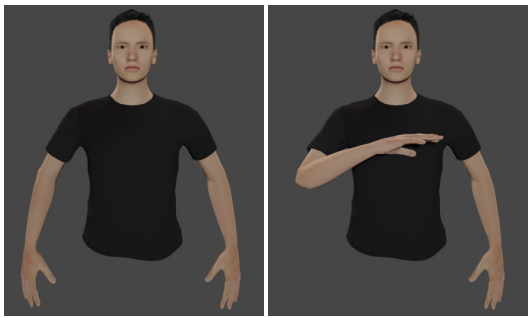

FIGURE 11. Left: initial and ending position. Right: middle position.

C. ANIMATION IMPROVEMENT

Just by registering the points in the different frames, the animation looks *robotic*, i.e. it does not reproduce a smooth and natural movement. The animation can be improved to solve this problem in two ways: adding a delazy movement and/or changing the interpolation type.

As mentioned earlier, to reach a position, it is necessary to move different joints and record the final position of all of them. As an example, if the right elbow, wrist and thumb are moved to a desired point, the position of all the bones is stored as a key frame. The delazy movement consists of moving random points forward or backward by one or two frames.

This change creates a more natural and smoother movement. This is because humans do not move each bone involved in a trajectory at the same instant of time, but instead move them out of phase with each other. Thus, this modification was applied to each movement.

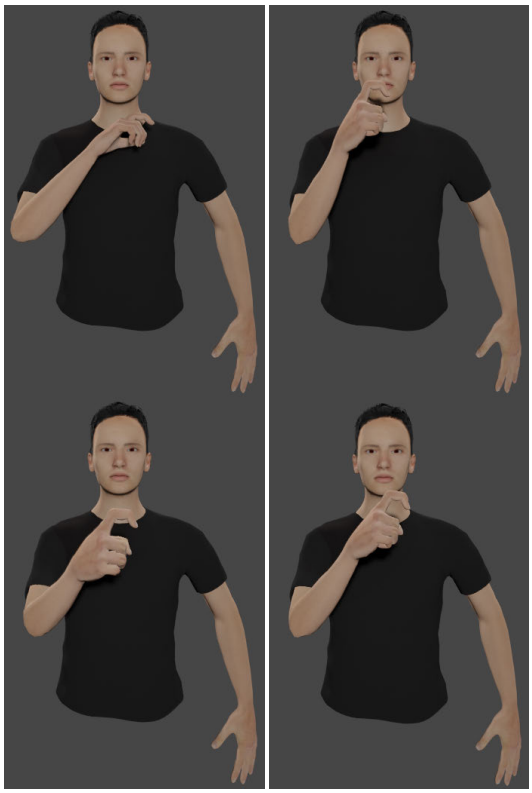

FIGURE 12. Example of some characteristic positions for letter *x*.

The second way to improve the animation was to change the default interpolation. Note that Blender calls *ease* to the smoothness of the interpolation. So the different types of interpolation are:

- Constant: Changes instantly from one frame to the next without curvature.
- Linear: Straight line interpolation without ease.
- Bezier (default): Smooth interpolation between point A and B but with no control over the ease.
- Sinusoidal: First degree polynomial, nearly a straight line.
- Quadratic: Quadratic polynomial.
- Cubic: Cubic polynomial.
- Quartic: Quartic polynomial.
- Quintic: Quintic polynomial.
- Exponential: Exponential curvature.
- Circular: Produces a rounded transition.

Note that the polynomial interpolations tend to be closer to the start or end point as the degree of the polynomial increases.

Furthermore, three different smoothness modes can be selected for the polynomial, exponential and circular interpolations:

- Ease In: The interpolation is close to the start frame. That is, the movement stays at the start point for a longer time.

- Ease Out: The interpolation is near to the last frame. This means that the movement stays at the end point for more time.
- Ease In and Ease Out: The interpolation is equal. The movement generates a linear transition, staying near the start and end point for the same amount of time.

With these configurations, there were four parts across all the sign where the interpolation method was changed:

1) Neutral position to middle position: Sinusoidal interpolation with ease in, used to stay in the neutral position longer.
2) Middle position to first signing position: Quadratic interpolation with ease out, used to reach the signing point and start signing faster but more smoothly.
3) Last signing position to middle position: Quadratic interpolation with ease in, used to stay in the signing point as long as possible.
4) Middle position to neutral position: Sinusoidal interpolation with ease out, used to reach the neutral position a bit faster.

For the points that make up the sign, there was no specific rule: sometimes the interpolation was changed to sinusoidal, quadratic, or even linear, and in other cases it remained the default (bezier). Nevertheless, there were some similar cases, i.e. in signs with repetitive movements like *ll*, *rr*, *ñ*, *v*, *w* or *y* a quadratic interpolation was used to accelerate the movement.

Finally, with these little improvements in the animation, all the signs composing the alphabet were recorded and saved in a video format. All these videos form a database that will be used by the complete application to fingerspell the word typed by the user.

V. THE APPLICATION

As illustrated in Figure 13, the system's workflow is as follows:

- when the user types a word, the avatar functionality is invoked so that the letters composing the word are recovered from the database and used to animate the virtual avatar.
- when the user fingerspells a word, it is analyzed by the recognition engine and the signed letters are displayed as text on the screen.

Thus, depending on the user's input, the system invokes the appropriate functionality.

With the aim to evaluate the application's performance and usability, a visual environment was designed.

In the case of spoken Spanish to LSE transcription, the environment allows the user to type what he/she wants to communicate, and the virtual avatar is responsible for signing it letter by letter. Figure 14 shows an example of this environment, while a video demo is available at https://youtu.be/uBssQFytV7o.

In a similar way, a demonstration environment has been created for the LSE to Spanish recognition (Figure 15) to showcase the system's capabilities. The demonstration shows the signing user in the bottom left and right sections, with and without the detected skeleton, respectively. The skeleton is color-coded in the upper left section, representing the input to the Neural Network model, and the letter being recognized by the system is displayed in the upper right position. A video demo is available at https://youtu.be/9Qme2KIvwro. It is important to note that, while the demonstration displays the complete process, the system's output is only the recognized letter as depicted on the right side of Figure 13.

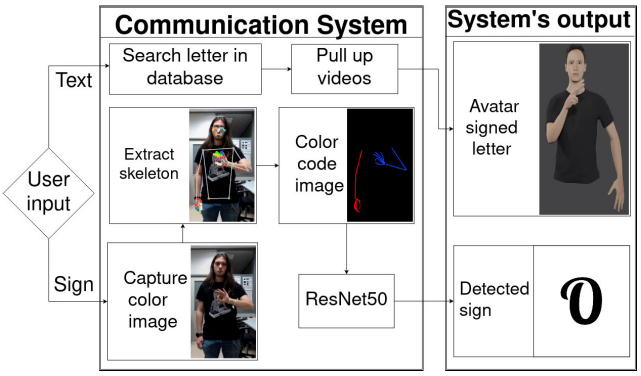

FIGURE 13. System's workflow.

FIGURE 14. Visual environment for spoken Spanish to LSE interpretation.

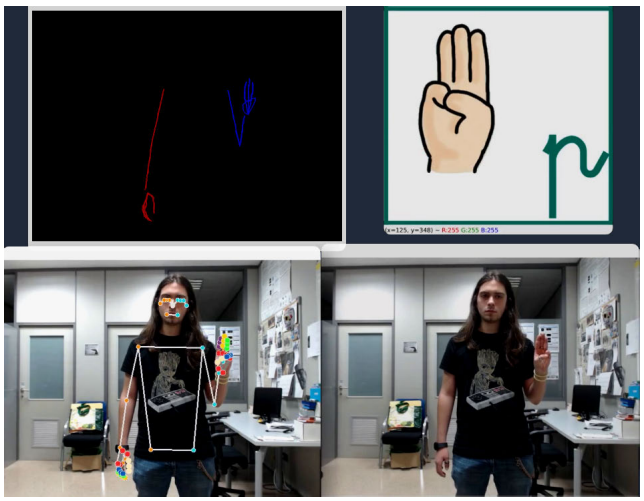

FIGURE 15. Created environment for LSE to Spanish interpretation.

TABLE 5. Example of prediction when post-processing is used.

Letter	1st	2nd	3rd	4th	5th	6th	7th	Mode
b	d	b	b	b	g	q	a	b
c	c	c	q	c	a	c	c	c
g	ñ	g	g	k	b	g	b	g
i	i	i	a	i	a	a	a	a
j	a	a	a	j	j	j	j	j
l	j	l	l	l	l	i	i	l
n	m	n	n	m	ñ	ñ	m	m
o	o	s	o	o	s	s	o	o
p	p	a	a	p	p	p	p	p
q	b	b	b	q	q	q	q	q
s	k	s	t	t	t	t	t	t

To improve the accuracy, an additional post-processing step was implemented in the LSE to spoken Spanish recognition engine. To achieve this, the system considers seven consecutive predictions for each letter before displaying the result. The mode of these predictions is then calculated and displayed, eliminating some erroneous values (see Table 5). The number seven was deliberately chosen to ensure an odd value that would avoid any duplication in the mode and maintain a speed of 10 FPS on an Intel i7 7700HQ CPU at 3.8 GHz with an NVIDIA GeForce GTX 1050 Mobile GPU. This post-processing step improved the accuracy from 78.81% to 79.96%.

Finally, the application was also evaluated in a real scenario. In this sense, as stated in [45], it is necessary to include deaf people, Sign Language linguistics, interpreters or experts in any areas related with Sign Language in the development and testing of Sign Language recognizers. So, to check the viability of the system, a professional interpreter of LSE tested the application and provides us with the required feedback to improve it.

In her opinion, although this project is in a first stage, its development could be really useful for the deaf community. Moreover, according to her feedback, the avatar was improved to make it more realistic and to make it sign at a speed more understandable to a Sign Language user.

She encouraged us to continue our research and enrich the application by adding more words and complex syntax in order to achieve a real translator.

VI. CONCLUSION

Hearing loss is a condition that can lead to social exclusion in everyday situations, so it is important to find new ways to overcome this scenario. Although some hardware solutions are available, there is a part of deaf population that use Sign Language as their form of communication.

This paper presents a low-cost application to facilitate the communication process between hearing and deaf people. In particular, in this early version, the alphabet is used. Note that, although the alphabet is used just in some situations, it is an essential and foundational part of any Sign Language, working as a link between deaf and hearing people, and even between deaf people coming from different communities or geographical points. So, it can be considered as a common, unified and independent communication system, being one of the most important steps in learning this language. Therefore, the presented application is able to display on screen what word is fingerspelled and to fingerspel the word or sentence typed by the user.

Regarding the LSE to spoken Spanish engine, different deep learning architectures were trained and evaluated to properly recognize the complete Spanish Sign Language alphabet. This resulted in a top accuracy of 78.81% on the test set. This accuracy was improved by means of a post-processing technique. So, the predicted letter is obtained as a mode of seven consecutive predictions. This process reduced the erroneous predictions and increased the system's accuracy up to 79.96%.

In addition, this comparative evaluation highlighted the importance of the spatial dimension over the temporal one in the case of the LSE alphabet. This fact is because, although some letters require movement, the position of the hands in the space is quite different from one letter to another. Moreover, there are more *static* signs than *in-motion* ones. In terms of erroneous recognition, the most recognition failures lie in the pair of letters whose only difference is movement: *i-y*, *l-ll*, *n-ñ*, *r-rr*, and *u-v*.

It is worth noting that we also expanded our previous dataset from 8, 000 images to nearly 28, 000 by increasing the number of participants, what allowed the system to achieve a better generalization.

For the virtual avatar, the whole alphabet was recreated and stored as animations in a database.

In order to verify the system's performance in real-life scenarios, an application was developed. This application was designed to choose the appropriate functionality according to the user's input. That is, if the user types a text word, the virtual avatar is invoked and signs that word letter by letter. on the contrary, if the user fingerspells a word, the recognition engine is called and the signed word is displayed on the screen.

Finally, the application's usefulness and performace were evaluated by a professional Spanish Sign Language interpreter. Although she evaluated the application positively, she suggested some changes to increase the naturalness of the signing through the avatar, that has been integrated into the application. In addition, she considers that the application can help deaf people in their daily communications, although it is necessary to add signs corresponding to the considered situations (e.g. shopping, healthcare, restaurants) and Sign Language grammar rules to make the application useful for this community.

Despite the promising results, the application has its limitations. Firstly, it only works with the alphabet. Despite the manual alphabet allowed us to analyze architectures for sign recognition, it is necessary to include more signs and grammar rules to successfully achieve a real translator between LSE and spoken Spanish.

Another limitation of this application is the reached accuracy, 79.96%. Although it is a good accuracy, it should be improved in order to get a useful and usable application.

Note that this is an early step towards breaking down communication barriers, but there is still a lot of work to be done. Firstly, information about the user's face needs to be integrated into the system in order to cover the different aspects involved in signing such as hand/finger's movement, signing space, facial expression, lip movement, and the use of one or two hands. Additionally, information relating to grammar and the lexicon will also be added to the system with the aim to achieve a useful application. In this sense, we have started a collaboration with FESORD, a Spanish association of deaf people, to participate in the development and test of the application.

VII. DECLARATION OF COMPETING INTEREST

The authors have no competing interests to declare that are relevant to the content of this article

VIII. ETHICAL APPROVAL

This study does not contain any studies with human or animal subjects performed by any of the authors.

ACKNOWLEDGMENT

Special thanks to Luz María Martínez Pérez, Spanish Sign Language interpreter, who tested the system and helped in its improvement.

REFERENCES

[1] (2018). *Spanish National Institute of Statistics (Instituto Nacional de Estadística (INE))*. Accessed: Jul. 5, 2023. [Online]. Available: https://www.ine.es/jaxi/Datos.htm?path=/t15/p418/a2008/hogares/p01/odulo1/l0/&file=01009.px#!tabs-grafico

[2] N. P. Aguado and M. D. P. Fernáandez-Viade, "The deaf literacy (DEAFLI): A European project for young and adult deaf people e-learning," *J. Health Sci.*, vol. 5, no. 2, pp. 73–80, Apr. 2017.

[3] I. M. Muñoz-Baell, M. T. Ruiz-Cantero, C. Á. Dardet, E. Ferreiro-Lago, and E. Aroca-Fernández, "Comunidades sordas: ¿ pacientes o ciudadanas?" *Gaceta Sanitaria*, vol. 25, no. 1, pp. 72–78, 2011. [Online]. Available: http://scielo.isciii.es/scielo.php?script=sci_arttext&pid=S0213-91112011000100012&nrm=iso

[4] S. Saha, (2022). *For the deaf Community, Sign Language Equals Rights*. Accessed: Jul. 2023. [Online]. Available: https://www.hrw.org/news/2022/09/23/deaf-community-sign-language-equals-rights

[5] D. J. Napoli, N. K. Mellon, J. K. Niparko, C. Rathmann, G. Mathur, T. Humphries, T. Handley, S. Scambler, and J. D. Lantos, "Should all deaf children learn sign language?" *Pediatrics*, vol. 136, no. 1, pp. 170–176, Jul. 2015, doi: 10.1542/peds.2014-1632.

[6] I. C. Báez-Montero and A. M. Fernández-Soneira, "Colours and numerals in Spanish Sign Language (LSE)," in *Semantic Fields in Sign Languages*. Berlin, Germany: De Gruyter Mouton, 2016, pp. 73–122, doi: 10.1515/9781501503429-003.

[7] M. B. Waldron and S. Kim, "Isolated ASL sign recognition system for deaf persons," *IEEE Trans. Rehabil. Eng.*, vol. 3, no. 3, pp. 261–271, Sep. 1995.

[8] M. W. Kadous, "Machine recognition of Auslan signs using PowerGloves: Towards large-lexicon recognition of sign language," in *Proc. Workshop Integr. Gesture Lang. Speech*, 1996, pp. 165–174.

[9] C. Vogler and D. Metaxas, "Adapting hidden Markov models for ASL recognition by using three-dimensional computer vision methods," in *Proc. IEEE Int. Conf. Syst., Man, Cybern.*, Oct. 1997, pp. 156–161.

[10] P. Lokhande, R. Prajapati, and S. Pansare, "Data gloves for sign language recognition system," in *Proc. IJCA Nat. Conf. Emerg. Trends Adv. Commun. Technol. (NCETACT)*, Jun. 2015, pp. 11–14.

[11] M. S. Kute and G. Chinchole, "Sign language to speech conversion device," in *Proc. ICRTIT*, 2020, pp. 1–6.

[12] Y. Pei, S. Biswas, D. S. Fussell, and K. Pingali, "An elementary introduction to Kalman filtering," *Commun. ACM*, vol. 62, no. 11, pp. 122–133, Oct. 2019.

[13] J. Han, G. Awad, and A. Sutherland, "Automatic skin segmentation and tracking in sign language recognition," *IET Comput. Vis.*, vol. 3, pp. 24–35, Mar. 2009.

[14] E.-J. Holden and R. Owens, "Visual sign language recognition," in *Multi-Image Analysis*, vol. 2032. Berlin, Germany: Springer, 2001, pp. 270–287, doi: 10.1007/3-540-45134-X_20.

[15] T. Starner and A. Pentland, "Real-time American sign language recognition from video using hidden Markov models," in *Motion-Based Recognition*. Dordrecht, The Netherlands: Springer, 1997, pp. 227–243.

[16] P. Kumar, R. Saini, P. P. Roy, and D. P. Dogra, "A position and rotation invariant framework for sign language recognition (SLR) using Kinect," *Multimedia Tools Appl.*, vol. 77, no. 7, pp. 8823–8846, Apr. 2018.

[17] L. Rabiner and B. Juang, "An introduction to hidden Markov models," *IEEE ASSP Mag.*, vol. ASSPM-3, no. 1, pp. 4–16, Jan. 1986.

[18] R. Rastgoo, K. Kiani, and S. Escalera, "Sign language recognition: A deep survey," *Expert Syst. Appl.*, vol. 164, Feb. 2021, Art. no. 113794. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S095741742030614X

[19] M. M. Kamruzzaman, "Arabic sign language recognition and generating Arabic speech using convolutional neural network," *Wireless Commun. Mobile Comput.*, vol. 2020, May 2020, Art. no. 3685614.

[20] L. Pigou, S. Dieleman, P.-J. Kindermans, and B. Schrauwen, "Sign language recognition using convolutional neural networks," in *Computer Vision ECCV Workshops*. Cham, Switzerland: Springer, 2015, pp. 572–578.

[21] B. Triwijoyo, A. Karnaen, and A. Adil, "Deep learning approach for sign language recognition," *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, vol. 9, pp. 12–21, Mar. 2023.

[22] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," 2015, *arXiv:1511.08458*.

[23] A. Nagaraj. (2022). *ASL Alphabet*. Accessed: Jul. 2023. [Online]. Available: https://www.kaggle.com/datasets/nkm9011/asl-alphabet-with-extension-of-akash-nagaraj

[24] I. Rodríguez-Moreno, J. M. Martínez-Otzeta, and B. Sierra, "A hierarchical approach for Spanish sign language recognition: From weak classification to robust recognition system," in *Intelligent Systems and Applications*, K. Arai, Ed. Cham, Switzerland: Springer, 2023, pp. 37–53.

[25] L. Hu, J. Li, J. Zhang, Q. Wang, B. Zhang, and P. Tan, "A speech-driven sign language avatar animation system for hearing impaired applications," in *Proc. 31st Int. Joint Conf. Artif. Intell.*, Jul. 2022, pp. 5912–5915, doi: 10.24963/ijcai.2022/852.

[26] Z. Bhatti, F. Muhammad, H. A. M. Malik, M. Hussain, H. Chandio, S. Channa, and Z. Mahar, "Text to animation for sign language of Urdu and Sindhi," *iKSP J. Emerg. Trends Basic Appl. Sci.*, vol. 1, no. 1, pp. 8–14, Feb. 2021. [Online]. Available: https://iksp.org/journals/index.php/ijetbas/article/view/60

[27] *SiMAX Project*. Accessed: Jul. 2023. [Online]. Available: https://simax.media/

[28] V. Lombardo, C. Battaglino, R. Damiano, and F. Nunnari, "An avatar-based interface for the Italian sign language," in *Proc. Int. Conf. Complex, Intell., Softw. Intensive Syst.*, Jun. 2011, pp. 589–594.

[29] L. Vera, I. Coma, J. Campos, B. Martínez, and M. Fernández, "Virtual avatars signing in real time for deaf students," in *Proc. GRAPP/IVAPP*, 2013, pp. 261–266.

[30] V. López-Ludeña, R. San-Segundo, C. G. Morcillo, J. C. López, and J. M. P. Muñoz, "Increasing adaptability of a speech into sign language translation system," *Expert Syst. Appl.*, vol. 40, no. 4, pp. 1312–1322, Mar. 2013.

[31] E. Martinez-Martin and F. Morillas-Espejo, "Deep learning techniques for Spanish sign language interpretation," *Comput. Intell. Neurosci.*, vol. 2021, Jun. 2021, Art. no. 5532580.

[32] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: Realtime multi-person 2D pose estimation using part affinity fields," 2018, *arXiv:1812.08008*.

[33] C. Lugaresi, J. Tang, H. Nash, C. McClanahan, E. Uboweja, M. Hays, F. Zhang, C.-L. Chang, M. G. Yong, J. Lee, W.-T. Chang, W. Hua, M. Georg, and M. Grundmann, "MediaPipe: A framework for building perception pipelines," 2019, *arXiv:1906.08172*.

[34] J.-L. Chung, L.-Y. Ong, and M.-C. Leow, "Comparative analysis of skeleton-based human pose estimation," *Future Internet*, vol. 14, no. 12, p. 380, Dec. 2022. [Online]. Available: https://www.mdpi.com/1999-5903/14/12/380
[35] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.
[36] A. Sherstinsky, "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) network," *Phys. D, Nonlinear Phenomena*, vol. 404, Mar. 2020, Art. no. 132306, doi: 10.1016%2Fj.physd.2019.132306.
[37] F. Morillas-Espejo and E. Martinez-Martin, "Sign4all: A real-time platform for Spanish sign language interpretation," *Expert Syst. Appl.*, to be published.
[38] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015, *arXiv:1512.03385*.
[39] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2017, *arXiv:1412.6980*.
[40] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
[41] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. 1997.
[42] *Uncanny Valley explanation*. Accessed: Jul. 2023. [Online]. Available: https://en.wikipedia.org/wiki/Uncanny_valley,
[43] *Blender's Page*. Accessed: Jul. 2023. [Online]. Available: https://www.blender.org/
[44] *MakeHuman Official Site*. Accessed: Jul. 2023. http://www.makehumancommunity.org/
[45] D. Bragg, O. Koller, M. Bellard, L. Berke, P. Boudreault, A. Braffort, N. Caselli, M. Huenerfauth, H. Kacorri, T. Verhoef, C. Vogler, and M. R. Morris, "Sign language recognition, generation, and translation: An interdisciplinary perspective," in *Proc. 21st Int. ACM SIGACCESS Conf. Comput. Accessibility*, Oct. 2019, p. 16, doi: 10.1145/3308561.3353774.

FRANCISCO MORILLAS-ESPEJO (Member, IEEE) received the degree in robotics engineering and the master's degree in automation and robotics from the University of Alicante, where he is currently pursuing the Ph.D. degree. He has published one JCR-indexed article. He has participated in some projects related with the use of computer vision and artificial intelligence.

ESTER MARTINEZ-MARTIN (Senior Member, IEEE) received the first master's degree in secondary education, vocational training, and language teaching, the second master's degree in mobile and video games programming, and the Ph.D. degree in engineering (robotics) from Jaume—I University, in 2013, 2017, and 2011, respectively. She has been a computer science engineer, since 2004. She is currently an Associate Professor with the University of Alicante. She has published 12 JCR-indexed articles, several articles in Scopus-indexed journals, a research book (Springer), several book chapters, three research books (as a co-editor), and more than 25 articles in congresses, both national and international. She has participated in several research projects of community, national, and European nature. She has also done four research stays in prestigious foreign centers, including Universitã degli Studi di Genova (Prof. Silvio Sabatini), Sungkyunkwan University (Prof. Suthan Lee), Universidade do Minho (Prof. Paulo Novais), and Technische Universitat (TU) Wien (Prof. Markus Vincze), all of which were financed in competitive public calls. Her research interest includes the use of vision in robotic tasks, such as object detection and action recognition. She is an AERFAI Member. She was a member of the program committee and organization committee in several national and international conferences. She has been the Organization Chair of the IEEE-RAS Summer School on Experimental Methodology, Performance Evaluation and Benchmarking in Robotics, in 2015; the 13th International Conference on the Simulation of Adaptive Behavior (SAB 2014); and the 12th International AERFAI/UJI Robotics School on Perceptual Robotics for Humanoids (IURS 2012). She was also a Co-Organizer of some tutorials in international conferences, including HRI, in 2017; ICINCO, in 2014; IAS-13, in 2014; and IEEE RO-MAN, in 2013. She is a regular reviewer of JCR journals and international congresses.