big data and cognitive computing

MDPI

Article

Sign-to-Text Translation from Panamanian Sign Language to Spanish in Continuous Capture Mode with Deep Neural Networks

Alvaro A. Teran-Quezada [1], Victor Lopez-Cabrera [1], Jose Carlos Rangel [1,2,3] and Javier E. Sanchez-Galan [1,2,3,*]

[1] Facultad de Ingeniería de Sistemas Computacionales, Universidad Tecnológica de Panamá (UTP), El Dorado, Panama City P.O. Box 0819-07289, Panama; alvaro.teran@utp.ac.pa (A.A.T.-Q.); victor.lopez@utp.ac.pa (V.L.-C.); jose.rangel@utp.ac.pa (J.C.R.)
[2] Centro de Estudios Multidisciplinarios en Ciencias, Ingeniería y Tecnología (CEMCIT AIP), Universidad Tecnológica de Panamá (UTP), El Dorado, Panama City P.O. Box 0819-07289, Panama
[3] Sistema Nacional de Investigación, SENACYT, Edificio 205, Ciudad del Saber, Clayton, Balboa, Panama City P.O. Box 0816-02852, Panama
* Correspondence: javier.sanchezgalan@utp.ac.pa; Tel.: +507-560-3933

Abstract: Convolutional neural networks (CNN) have provided great advances for the task of sign language recognition (SLR). However, recurrent neural networks (RNN) in the form of long–short-term memory (LSTM) have become a means for providing solutions to problems involving sequential data. This research proposes the development of a sign language translation system that converts Panamanian Sign Language (PSL) signs into text in Spanish using an LSTM model that, among many things, makes it possible to work with non-static signs (as sequential data). The deep learning model presented focuses on action detection, in this case, the execution of the signs. This involves processing in a precise manner the frames in which a sign language gesture is made. The proposal is a holistic solution that considers, in addition to the seeking of the hands of the speaker, the face and pose determinants. These were added due to the fact that when communicating through sign languages, other visual characteristics matter beyond hand gestures. For the training of this system, a data set of 330 videos (of 30 frames each) for five possible classes (different signs considered) was created. The model was tested having an accuracy of 98.8%, making this a valuable base system for effective communication between PSL users and Spanish speakers. In conclusion, this work provides an improvement of the state of the art for PSL–Spanish translation by using the possibilities of translatable signs via deep learning.

Keywords: hand gesture recognition; sign language recognition; convolutional neural networks; object detection; transfer learning; machine learning; deep learning; convolutional neural network; recurrent neural network

Citation: Teran-Quezada, A.A.; Lopez-Cabrera, V.; Rangel, J.C.; Sanchez-Galan, J.E. Sign-to-Text Translation from Panamanian Sign Language to Spanish in Continuous Capture Mode with Deep Neural Networks. *Big Data Cogn. Comput.* **2024**, *8*, 25. https://doi.org/10.3390/bdcc8030025

Academic Editors: Moulay A. Akhloufi, Robail Yasrab and Md Mostafa Kamal Sarker

Received: 16 September 2023
Revised: 2 February 2024
Accepted: 21 February 2024
Published: 26 February 2024

1. Introduction

Modern information and communication technologies (ICT) have solved a myriad of problems. For instance, long-distance communication and, more importantly, the communication between two people who do not speak the same language can both be experienced in real-time. However, there is still a debt with regards to the inclusion of people with disabilities. This is especially true when it comes to hearing impairments and for non-English speakers. One aspect that could potentially make a difference for this technologically underprivileged community would be a system capable of recognizing signs from a visual input and presenting a text with the corresponding translation into the user's respective language.

There is a wealth of information and studies about sign recognition developments with machine learning methods [1] and especially with deep convolutional neural networks (DCNN) [2–4] . However, implementations have seldom been made with recurrent neural networks (RNN), in comparison to other structures. Our hypothesis and motivation for

making a development under this approach is quite simple. It boils down to the fact that although it is true that CNNs are notoriously good at processing visual data (such as the image of a sign), they have limitations in processing groups of data as a single element. On the other hand. RNNs specialize in recognizing the sequential data set and are geared to find the characteristics of this kind of data. This type of network facilitates the processing of groups of visual data as a single element of the sign. This is particularly true for sets of poses and hand signs that correspond to a specific meaning.

In this work, the development of a system capable of recognizing signs from the Panamanian Sign Language (PSL) with deep neural networks in continuous capture is proposed. The main objective is to have PSL signs translated into their Spanish written text format. This will help establish an effective communication channel between people with hearing disabilities and those who are not PSL users.

For this, the proposed solution first goes through a general design of the project, the development of a model that translates static signs from an object detector, followed by a design of the model for the translation of dynamic signs, which implies the construction of an action detector and the development of the model itself. This will be the center of the (PSL–text) sign translation system, which must be optimized in terms of computational resources and precision during recognition.

2. Related Work

2.1. SLR Using Deep Learning Methods

The field of computer vision has made significant progress given a mass amount of research and practical projects. The publications that are most related to this project involve the use of the convolutional neural network, more precisely, focusing on sign translation (sign language recognition or SLR). SLR is usually associated with translation between American Sign Language (the one used in the United States and Canada) and English, due to the research volume coming from the countries in which these languages coexist.

Inspecting the available literature, at least two ASL solutions caught our attention: a hand gesture recognition model trained with a data set created for that specific purpose [5] and a digital read number-sign classifier, where special preprocessing is implemented to enhance accuracy in translation [6]. In both these cases, CNN architectures are used for sign translation, something that despite having been tried on several occasions, continues to prove its applicability due to the marked need of the ASL users.

Deep learning has revolutionized SLR, particularly for isolated signs. Convolutional neural networks (CNNs) have achieved significant accuracy, as demonstrated by the work of Marjusalinah et al. (2021) on finger spelling recognition with 99% accuracy [7]. The recent advancements explore recurrent neural networks (RNNs) and transformers to capture the temporal dynamics of continuous signing, like the work of Ariesta et al. (2018), who achieved a word error rate of approximately 90% accuracy for sentence-level recognition using a 3D CNN and LSTM combination [8]. However, challenges remain, including robustness to variations in sign styles, backgrounds, location, and characteristics that are unique to the signer [9,10].

2.2. SLR with Traditional Analysis Methods

Without a doubt, one can say that deep learning dominates and is the go-to method to analyze the data in SLR. However, there are other methods that play a crucial role in SLR. Efforts are still being made to attack this problem from an important variety of perspectives, and it turns out that CNNs are leading the advances in the field. In addition to this first type of neural network, there is the recurrent neural network. These have been presented to propose novel solutions to new problems while simultaneously acting as an alternative to the previously addressed problems (up to a certain point, at least). Some current works supported by RNNs are related to sentiment analysis (also known as opinion mining), an approach to natural language processing (NLP). In fact, there are projects that

work together with CNN [11] or time-series classification (an area within ML) and ML forecasting, as in [12].

It is worth mentioning the approaches that existed at the time or that have coexisted with those of deep learning (particularly with CNN). Among them, those of machine learning, such as the one evidenced in [13] that worked with ML, are based on models and characteristics according to the linguistic composition of the lexical signs, and others continue to be under the umbrella of artificial intelligence, such as the use of radio frequency sensors to generate synthetic data [14].

For instance, hidden Markov models (HMMs) offer efficient recognition for well-defined sign features, as is shown by Zhang et al. in their work on Chinese Sign Language recognition [15]. Traditional machine learning approaches, like support vector machines (SVMs), offer interpretability and efficiency for smaller data sets [16,17]. Finally, combining traditional methods with deep learning can lead to improved performance and robustness [18–20].

2.3. SLR for Other Languages (Beyond ASL)

Other than for ASL, there exists relevant documentation on SLR applied to different languages. There is, for instance, a sign language semantic translation system (from Arabic Sign Language) using a combination of ontologies and CNNs [21] and a proposed network architecture for translating signs (from Bangladeshi Sign Language) [22]. Additionally, there are other implementations oriented to French Sign Language (Belgian SL, BSL) using deep learning methods [23].

There is also a system based on sensors captured by a device through which data is collected to perform the translation (of Italian Sign Language signs) using automatic learning algorithms [24]. Three (3) classifiers were used, of which the one focused on artificial neural networks (ANN) had the best performance compared to those of support vector machines and K-nearest neighbors.

The research obviously extends beyond ASL to encompass diverse sign languages like Arabic, Australian, Indian, and Chinese. The variations in signing grammar, vocabulary, and cultural influences pose challenges. Deep learning can adapt well but requires language-specific training data and careful consideration of cultural nuances [25–27].

2.4. SLR with Varying Capturing Methods

SLR shares common ground with image captioning, object recognition, and object tracking. Deep learning techniques developed in these domains readily apply to SLR, particularly in areas like hand pose estimation and sign segmentation [28,29].

Conversely, advancements in SLR can benefit these related fields by providing novel approaches to gesture recognition and action understanding in video data [30].

2.5. Development in Panamanian Sign Languages

In the specific case of Panamanian Sign Language, a number of projects are worth mentioning. There exists a compilation book for supporting the PSL learning process [31], a website for a similar purpose [32], a robotic hand as an alternative way to achieve inclusion through acquisition of PSL [33], and a mobile application with documentation and vocabulary from PSL [34]. Furthermore, the authors have worked on the *EnSenias* web platform with a series of translations of concepts in Spanish to their equivalent in PSL [35–37]. This tool is geared to mitigate the communicative problem between the hearing impaired and the rest of the population, with over 1250 signs.

In addition, Bodmer [38] presented a CNN-based system capable of translating signs corresponding to vowels in PSL with a high level of precision. Finally, our group recently presented a work geared to educational settings, in which a hand gesture recognizer is trained to be able to identify signs and numbers in arithmetic operations [39].

3. Methods

The proposed solution in this project is based on the Mediapipe platform. Specifically, on the Mediapipe Holistic [40] platform. The topologies selected for our task were hands [41], face [42] and pose [43], mostly due to the need to consider the semantic loaded parts of the body, along with important supporting factors, in addition to the main elements.

These topologies consist of several models working together. For instance, the hand tracking solution has a palm detector, a hand landmark model and a gesture recognizer that identify key-points, track them through time, and detect the gesture [44]. The face detection and mesh is achieved with augmented reality (AR) overlaying digital information (the landmarks) on top of the face [45]. The (human) body pose tracking accurately localizes landmarks of the body from a single frame [46]. All of these independent solutions perform in real-time, as the one presented, and are used to detect signs from the PSL.

The overall methodology used for T=the translation of sign-to-PSL is shown in Figure 1.

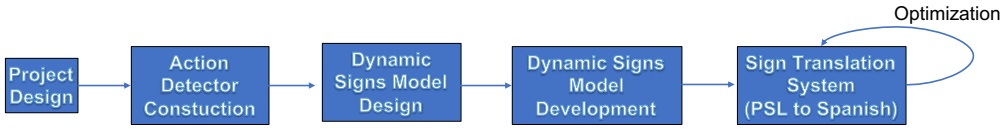

Figure 1. Methodology Diagram based on Action Detection for Sign Language Recognition.

First, the project, including all the topologies from Mediapipe, was selected. Then, an action detector was constructed. Next, a dynamic sign model is established and developed. Finally, the sign made and detected via this methodology was translated. There is also an optimization step, in which the process of translation is optimized; that is, the system is trained via feedback to be able to detect the sign with an acceptable accuracy.

3.1. Image Collection and Preprocessing

The image collection procedure was automated via a built-in webcam. Images and videos were collected one at a time, naming the files in a way that represents the sign or sign phrases captured.

For creating this data set, a Python script using OpenCV was written. The amount of data to be collected is directly related to the number of classes to be considered according to the experiment and responds to statistical heuristic strategy for a simplified decision regarding the sample size. The chosen method is the factor of the number of classes, which proposes an x number of elements for each class. This x should be a power of 10; in this case, it was $10^2 = 100$.

3.2. Development of the Dynamic Sign Module

For the development of the dynamic sign module, 4 key Python modules were used: (1) OpenCV was used to access the integrated camera and extract the keypoints, as well as for image processing; (2) Mediapipe (MP) was used to extract the points (keypoints) from the face, hands, and body; First, for training, and subsequently in execution for predictions, (3) Sklearn was used to calculate the evaluation metrics for the separation of the data in the training and testing sets and in stages for which the code is presented; and finally, Matplotlib was used for the visualization of the images.

3.2.1. Keypoint/Landmark Detection Using Mediapipe Holistic

First, the visualization of the reference points and their connections is configured; that is, checking if there are changes in the format parameters. Colors were established that easily differentiate the elements and that are compatible with a good number of backgrounds; the thickness of the lines and the size of the points respond to the convenience of presenting each component. It is noteworthy that different colors were configured to their hands to identify them more easily in the training, testing, and execution processes, since sometimes they are treated with normal visualizations and in others in mirror mode.

3.2.2. Capturing and Processing Images and Keypoints

An Opencv element of the video capture type was created and set to either 0 or 1 to capture via smartphone or via built-in camera, respectively. The minimum confidence values for detection and monitoring were set to 0.5 (50%). This value is vital to determine the specificity of the model. The capture, detection, and layout of the points and lines of the landmarks are carried out iteratively. All collected images are concatenated into a Numpy array. It will have "0" values if there is no input. These results are later analyzed to extract keypoint values.

3.2.3. Keypoint Value Collection for Training and Testing

Once all the images are collected, the resulting videos are created in order. That is, the first image of the first video is the first sign, then the following is the second, and so on for all signs. A slack time of two (2) seconds between video and video was programmed, which also has an effect between one sign and another, that is, the last video captured of a sign and first video to capture the next one.

3.2.4. Sequence Preprocessing and Creation of Labels and Features

The "Train Test Split" function of scikit-Learn (Sklearn) was called to make data partitions according to 90% for training and 10% for testing. The classes are assigned using the "to categorical" function from Keras. In general terms, this function turns a class vector into a binary class matrix. A dictionary labeling map is created to represent each of the signs ("Actions" for action detection). This map is used when the set of labels is created according to the sign.

All the collected keypoints, which can be seen now as sequences of keypoints), are structured into 90 arrays with 30 frames each and 1662 values representing the keypoints. This structure is later introduced into the RNN model.

Images and keypoints captured in this collection and described in this subsection were made publicly available and can be found in [47].

3.3. Model Definition

The LSTM recurrent network model was defined to have a 30×1662 input, followed by three LSTM layers of 64, 128, and 64 neurons, respectively. It was followed by two dense layers of 64 and 32 neurons, respectively. The ReLU activation function was used for every layer that required it, except for the output layer, which used the softmax activation function. The total number of parameters used by this network is 596,741 parameters.

The main idea is that a model is trained to be able to identify signs. Since the signs are identified according to the pre-established labels, the translation is achieved simply by presenting the text (the name of the class or label; in this case, a sign) through a visual output.

The construction of the sign recognition model started by using an action detector. Mediapipe Holistic was used for this purpose, having the different physical components assessed. This included identifying the points of interest and tracking them through time so an action was determined. These actions represent a sign in PSL. This detector was designed to be trained end-to-end, that is, from scratch.

3.4. Experimental Setting

The ready-to-use model would typically receive, as input, images where signs were made by the people in them, these signs being the objects to be detected in continuous capture. An overall view of the functioning of the system is depicted in Figure 2.

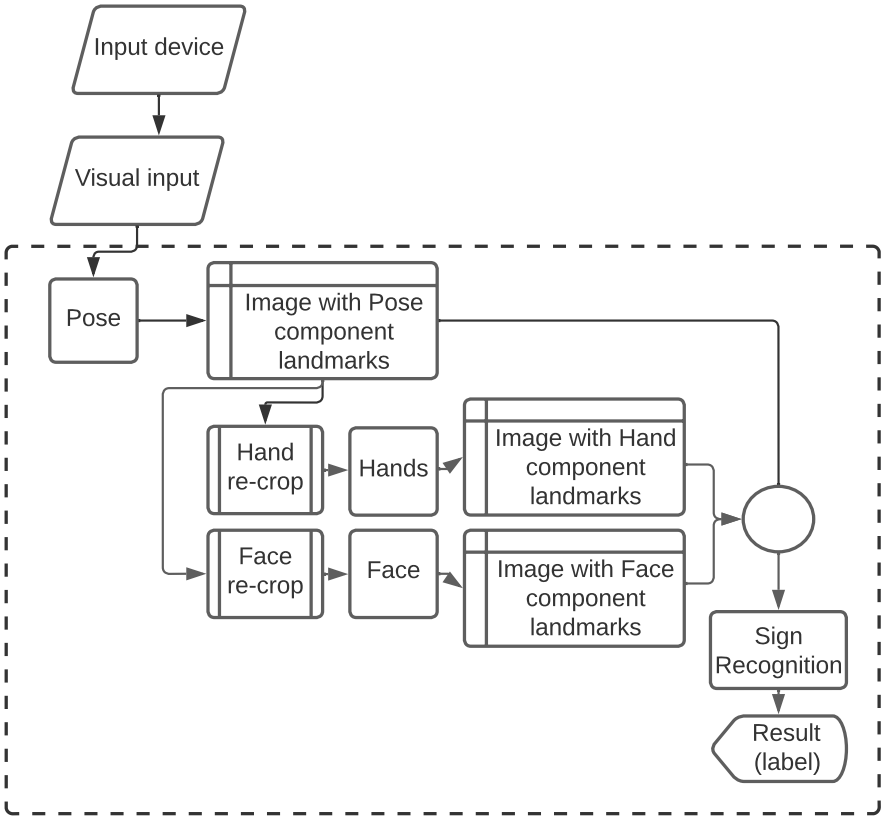

Figure 2. Overall View of the Proposed Recognition System.

The developed model is trained and works with PSL sign images, either in continuous capture (continuous shots) or when bringing a single image as an input. The system, once executed, begins to extract the keypoints of the person captured by the camera over time. With this, the sign translation is made when the model identifies one of the known signs.

The process used to achieve the task of translation involves the stages contemplated on the right side of Figure 2, with the model as the core of the system.

The results are obtained via a prediction of the model. The Numpy function argmax() is used to return the maximum values of the given axis; that is, the class it infers is more likely to correspond to the input.

The translation takes place as follows: the system captures frames one after the other with a very low delay. For each capture, the image inference process is carried out (it is expected that a person appears making a sign) from which percentages are obtained for each class; the label calculated as most probable is presented on the screen only if the reliability value (threshold) of 80% is reached. In other words, the sign of the PSL is translated in continuous capture into Spanish text.

3.5. Experiment Overview

Two experiments were devised to test the proposed system:

- Experiment #1: the objective of this experiment was to train a model that was able to translate dynamic signs with the deep neural network model. For this task, 5 signs were tested (Hola—Hello, Buenos días—Good morning, Estoy Bien—I am fine, Gracias—Thanks, ¿Cómo estás?—How are you?).
  For this experiment, a data set consisting of 625 videos (30 frames for each one) corresponding to the 5 classes (signs) were collected. Each sign was then captured 125 times, of which 20% (around 25 images per class) were used for validation.

- Experiment #2: the objective was to be able to assess the performance with a lesser number of classes. For this task, 3 signs were tested (Hola—Hello, Estoy Bien—I am fine, Gracias—Thanks).
  For this experiment, a reduced data set was considered. This data set consists of 375 videos (30 frames for each one) corresponding to the 3 classes (signs) remaining. Each sign was then captured 125 times, of which 15 images per class (12%) were used for validation.
  The hypothesis behind this experiment is that the model could have a much better performance in execution if the number of options to be considered is shortened, given that there is a shared probability between all the elements (possible options). Given this hypothesis, we aimed to improve performance by considering fewer classes.

Evaluation Metrics

This stage was completed using a quantitative analysis method common within machine learning: the confusion matrix. The metrics to be used for the evaluation of these models were the four (4) measurements that make up the confusion matrix: true positive (*TP*), false negative (*FN*), false positive (*FP*), and true negative (*TN*). A visual description of the confusion matrix can be seen in Table 1.

Table 1. Confusion matrix of Experiment #1's best scenario.

		Predicted Class	
		True	**False**
Observed Class	Positive	*TP*	*FN*
	Negative	*FP*	*TN*

Furthermore, the accuracy (Equation (1)), precision (Equation (2)), recall (Equation (3)), and F1 score (Equation (4)) of the models were calculated, using the following formulas:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (1)$$

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} = \frac{2 * TP}{2 * TP + FP + FN} \quad (4)$$

This was intended to validate the performance of the developed model to confirm both that the results were reliable and that they were going to be executed.

Confusion matrices were created for each model and for each class in Experiment #1 and #2, using sklearn [48], calling the multi-label version, thus returning confusion matrices for each class.

Moreover, the accuracy and loss of the models were calculated with the intention of validating the performance of the developed model, as well as to confirm both that the results were reliable and that there was the possibility of using the models in different settings.

4. Results

4.1. Image Acquisition

Figures 3 and 4 illustrate the process of the capture of each action (video of a sign). For instance, in Figure 3, the "hello" sign is shown. Figure 4 shows the captured "I'm fine" sign. Finally, Figure 5 shows the "Thanks" sign. Each sign was captured 125 times, of which 20% (25 images per class) were used for validation.

Figure 3. Sequence of the sign "Hello".

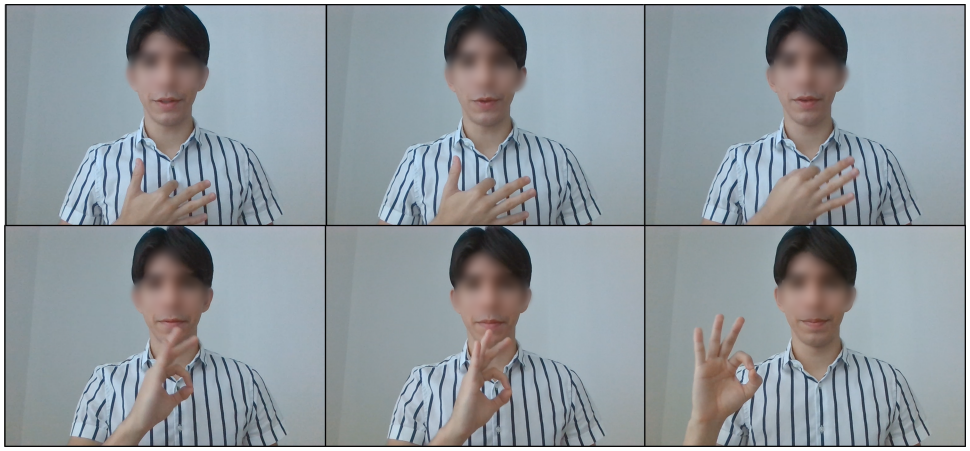

Figure 4. Sequence of the sign "I'm fine!".

Figure 5. Sequence of the sign "Thank you".

4.2. Detection Experiments

As can be seen in Table 2, both experiments provided appropriate results. Experiment #1 had a training time of 196.6 s, a loss of 0.222, and a categorical accuracy of 0.958, achieved in 160 epochs. In general, Experiment #2 performed faster than Experiment #1, with comparable results. The training time for Experiment #2 was 52.03 s, with a loss of 0.9948, and its categorical accuracy was 0.9879, achieved in 46 epochs.

Table 2. Resulting Values for the Experiments

Metric	Experiment #1	Experiment #2
Accuracy (global)	0.958	0.988
Accuracy (best class)	1.00	1.00
Accuracy (worst class)	0.96	1.00
Final Loss	0.222	0.995
Epochs	160	46
Training time (s)	196.6	38.88

4.3. Individual Class Performances for Experiment #1

This experiment was geared toward developing a model able to translate dynamic signs via a deep neural network model. For this task, five (5) signs were tested (Hello—Hola, Buenos dias—Good morning, Estoy Bien—I am fine, Gracias—Thanks, ¿Como estas?—How are you?).

Table 3 shows the resulting confusion matrix for the best predicted sign. One can see that the "Hello" sign has a higher *TP* value of 88%. It also shows perfect scores of 1.00 (100%) for all the metrics evaluated: precision, accuracy, and F1-score.

Table 3. Confusion matrix of Experiment #1's best scenario (with the number of elements and the percentage they represent).

		Predicted	
		True	**False**
Class "Hello"	Positive	22–88%	0–0%
	Negative	0–0%	3–12%

Table 4 shows the resulting confusion matrix for class "Good Morning" as the worst predicted sign, showing 96% in total for *TP* and *TN*. The precision was approximately 0.9524, accuracy was 0.9600, and F1 score had a value of 0.9756; therefore, over 95% in general.

Table 4. Confusion matrix of Experiment #1's worst scenario.

		Predicted	
		True	**False**
Class "Good morning"	Positive	20–80%	1–4%
	Negative	0–0%	4–16%

4.4. Individual Class Performances for Experiment #2

For this experiment, the goal was to create a similar model in Experiment #2 but with fewer categories. For this task, three (3) signs were tested (Hello—Hola, Estoy Bien—I am fine, Gracias—Thanks). Table 5 shows the resulting confusion matrix for each one of the "Hello" signs, the best class on Experiment #1, which had a precision, accuracy, and F1-score of 1.00 (100%).

Table 5. Confusion matrix of Experiment #2's best scenario.

		Predicted	
		True	**False**
Class "Hello"	Positive	7–46.67%	0–0%
	Negative	0–0%	8–53.33%

When looking at the performance per epoch, for accuracy per epoch (Figure 6A), one can see that there was a slow increase in value until 20 or so epochs were reached, until the best value was achieved in the reported 46 epochs. For the loss (Figure 6B), it seems to behave appropriately from the beginning, lowering in value until reaching a minimum in the final epoch.

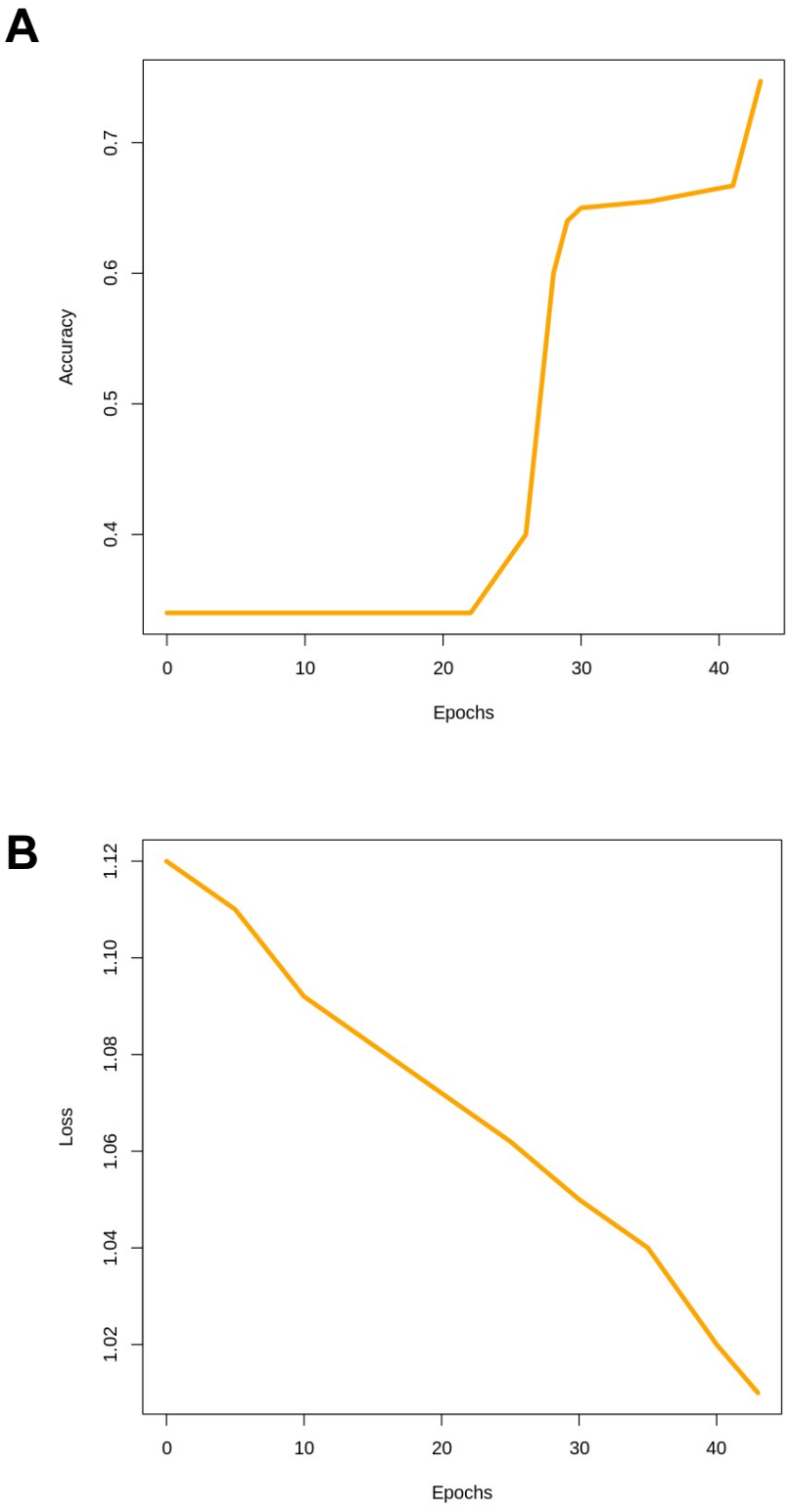

Figure 6. Accuracy per epoch (**A**), and Loss per epoch (**B**) for the Model in Experiment #2.

4.5. Real-Time Execution

When it comes to live recognition, it can be said that Experiment #2 was able to smoothly detect the signs for which it was trained. Figure 7 shows the model qualification of the gesture for the sign "Hello", while Figure 8 shows the continuation of the same execution, with the response "I'm fine!". The hypothesis behind Experiment #2 was correct. The model does have a good performance with fewer classes. This fact also proves the applicability and usefulness of the technology within SLR.

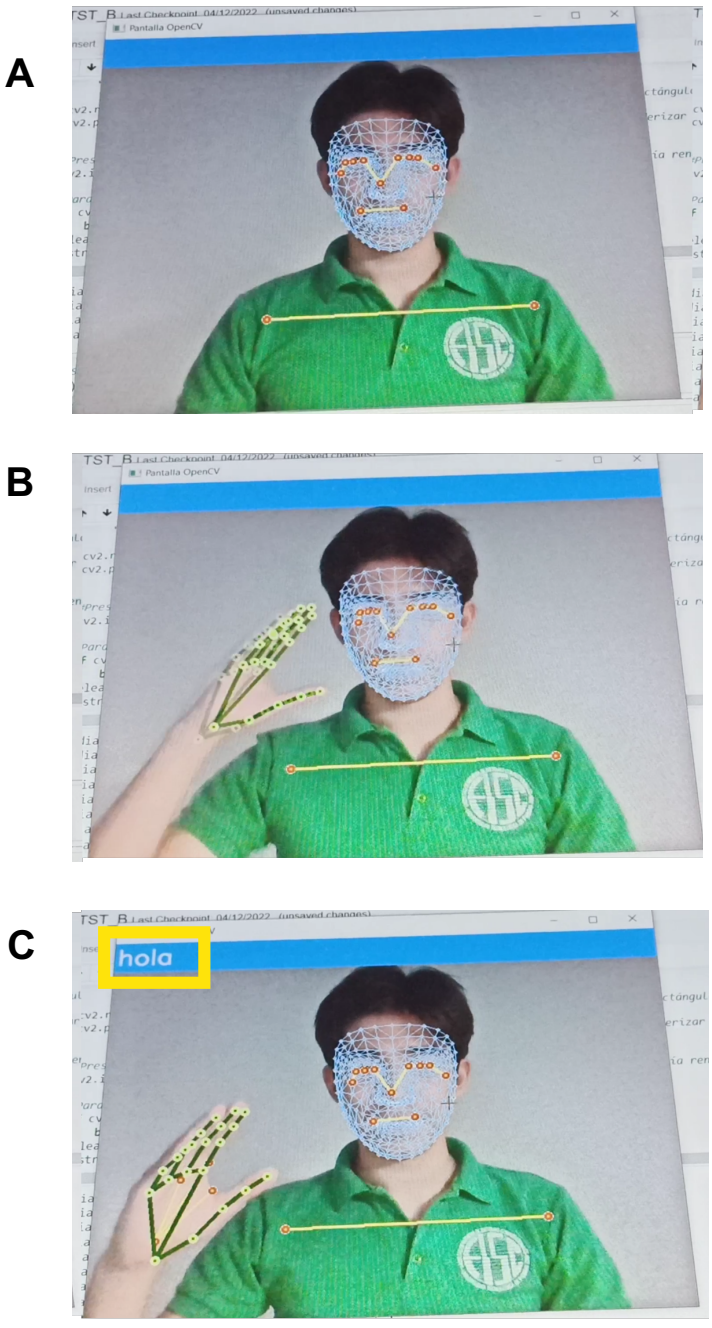

Figure 7. Recognition of the sign for "Hello". Panel (**A**) shows the start state with no sign, Panel (**B**) shows the start of the construction of the sign, Panel (**C**) shows the end of the sign for "Hello", more importantly the software is capable of showing the resulting recognised sign in the blue text box on the top of the picture.

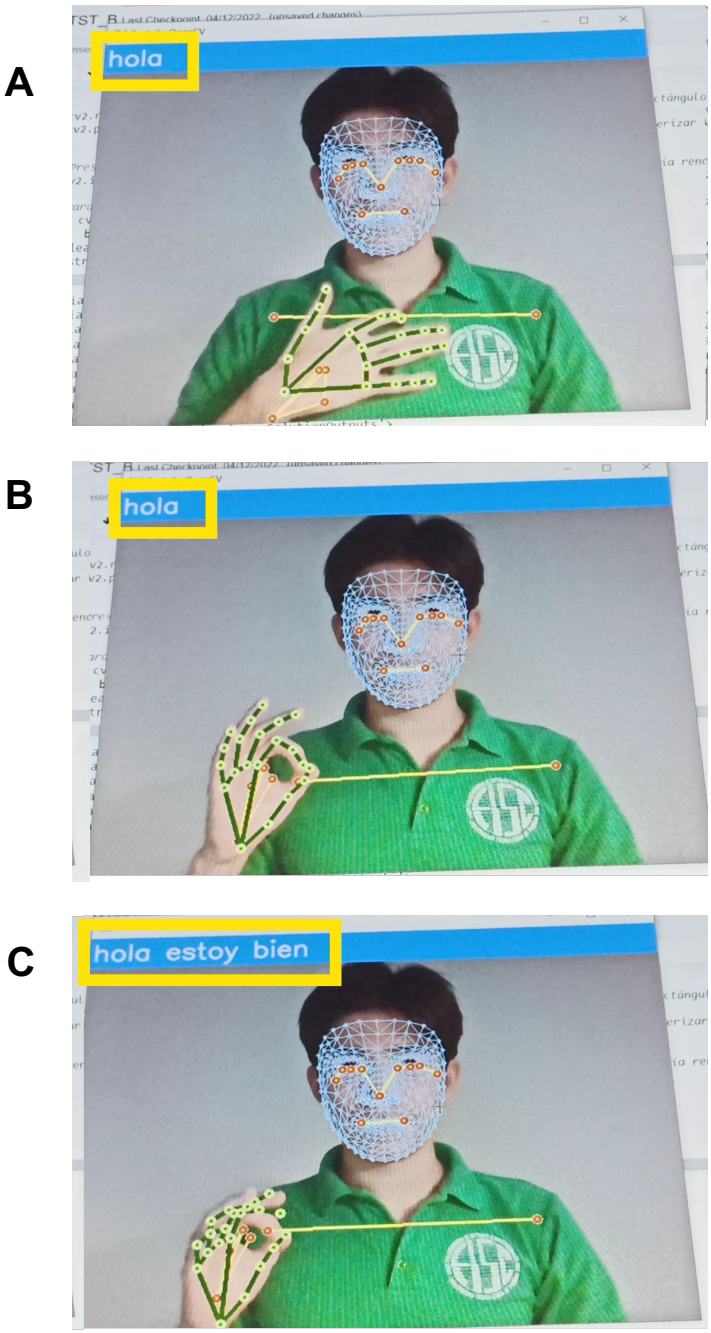

Figure 8. Recognition of the sign for "I'm fine!". Panel (**A**) shows the start of the sign, Panel (**B**) shows the middle of the sign, Panel (**C**) shows the end of the sign for "I'm fine", Notice that for Panels (**A**,**B**) the blue text box, still is showing the previous result shown in Figure 7. while in Panel (**C**) it changes to the correct sign, capture in continuos mode.

5. Discussion

The original idea behind the project was leveraging deep learning beyond CNNs to develop a sign language recognition (SLR) module that could translate motion signs. The use of recurrent neural networks (RNN) seemed natural, given that they excels in sequence data processing. The approach presented fundamental features for it to be considered in the future, such as a significantly low training time. When comparing Experiment #1 and Experiment #2, one can say that there is an overall performance improvement in the running model with fewer classes. The detection speed can be considered real-time, and the accuracy is high.

An important lesson from the experiments, either in Experiment #1 or Experiment #2, was learned when testing the recognition capacities when providing inputs captured at a different distance from the training set. The distances were doubled, reaching from 0.5 m (1.64 ft) to one meter (3.28 ft). This provided a clear limitation, although it was able to return correct results after some attempts. When training half a meter away, there was a noticeable difference between the values with which it was trained and those with which it was being tested. It is important to notice that these networks work by learning position values inside an image or frames. Moreover, a second reason points to the possibility that the architectures on which the solution is based may not be able to identify correctly the points of interest.

Another important lesson was learned when capturing sequences without noise. That is, an idle position is of utmost importance within video data, for instance, when starting the sign or after completing the gesture. This would mean having frames with similar positions for different classes, increasing the probability of a bad inference.

To understand the limits of detection of the system, two further variations of the experiments were designed. The first variation was implemented to study the relationship between the execution performance and varying the distance from 0.5 m to one (1) meter. It was found that the system is clearly limited by distance. The model being trained from 0.5 m does not generalize well. This could be attributed to the distance between the values with which it was trained and those with which it was being tested. A second reason indicates that there is the possibility that the topologies of the solution are not able to identify correctly the points of interest.

The second variation experiment included taking images in low lighting. For this case, the model was able to identify the signs (after translating them immediately) without major problems. However, if in any collection, a variation in distance and low light is combined, the performance is poor.

Looking forward, the developed model could be the base of a module for recognizing dynamic signs. While another model would focus on static sign recognition, a system with two independent modules might deal better with simultaneous translation of both types, because it will handle the data in which they are specialized. Furthermore, it can be the basis to make a text-to-speech/audio conversion module, making the system give an audio output in addition to the text one already shown.

The proposed system was built for the recognition of a few classes and with a small number of samples per class (25 and 15 in the testing class for each experiment, respectively). Experiment #2 used only three classes to train the final version of the model. Despite this, it had a great accuracy in recognition and a smooth execution process. Generative adversarial networks could be used for generating artificial sequences for training, basically creating frames and later sequences, as a data augmentation strategy [49,50].

The system described here presents a clear applicability to different use cases, for example, ASL–English. It is clear that there will have to be noticeably more classes to be recognized by the model. It is also evident that it is easily adaptable and it would require training with the actions corresponding to the signing in another sign language, along with labeling in a different language.

Moreover, this system can be extended by providing visual output to signs in another sign language, known as gesture-to-gesture systems [51], or even spoken output, as a sign language translator (SLT) [52]. This would make possible the communication between sign language users that do not know each other's SL system, for example, PSL–ASL.

Finally, the action detector could be adjusted to a separate task (not SLR), like detecting an indication from a vehicle traffic director hand signal. A differently-trained version of the model could be leveraged for sport analysis, gesture control, and so on.

6. Conclusions

Undoubtedly, machine learning has proven to be a means of solving countless problems of a considerable variety. In particular, deep learning has made its way to be a top

discipline in artificial intelligence, having represented changes in the state of the art of different processes, techniques, and areas. This subfield of ML has increased its relevance by having conceived DNN structures with interesting capabilities; and the support of elements, such as the availability of tools and resources provided by the scientific community, are the basis of new ones. Computer vision tasks, including SLR, have benefited from the growth of DL.

CNNs have continued to prove themselves useful for computer vision for several years due to their ability to perform well in tasks regarding visual data processing. Despite the superlative position of CNN structures in object detection, there are novel architectures such as U-nets and others that have proven to be valuable [53]. An object detector can be adapted to become the core of a sign-to-text translation system, but developed systems of this kind have presented an enormous challenge: the limitation of identifying only static signs, that is, only hand poses. This is where other structures can participate, and RNNs were the ones used this time, based on the need to process a sequence of images and not unrelated static images.

Dynamic signs are indeed actions. Gesturing in this type of sign implies motion, so it can be seen as different hand poses through time. Instead of a still image, like static signs, dynamic signs are studied like sequences of frames. Processing a sequence of frames, hence, an action, can be achieved with a combination of two DNNs: RNNs for sequence processing and CNNs for image processing.

From the above, a parallel can be seen between object detection and action detection. In action detection, RNNs bring the "memory" component, but how it is implemented may vary. When considering just the positional values of the hands of a person making the signs, a fairly good recognition could be achieved. While the hand component is the most important one in this use case, the consideration of other components might be convenient when developing a richer solution. The other components used, namely, pose and face, enhance the translation system through Holism. The pose component is advantageous when there are hands overlapping, since the model determines elements like wrists, then "understands" there are two hands. The face component is valuable because facial expressions are a relevant part of sign language, expressing emotions but also linguistic information.

This holistic approach proved itself useful when it came to SLR, especially for translating signs with motion, which has been a complicated problem in the area. In Experiment #2, the developed model had great results in validation and when being tested in execution. On the one hand, this means it is worth applying this technology to the sign recognition task. On the other hand, it means that the limitation regarding the number of classes that can be considered is yet to be overcome.

As pointed out previously, the limitations of this study are various. First, the environment is quite straightforward, the data are collected without noise or the intervention of other persons, and it basically is not for multiple signers at the same time. Next, iterations have more variations in brightness, number of persons, hand gestures (extension of the vocabulary), movements, facials, and backgrounds, all of which are challenges in the SLR community.

In terms of noise or variability, all the signs were collected with one experienced signer, and the model does not account for PSL students, and it does not provide corrections for poorly described signs. We understand that optimization in two aspects, (1) idle positions between signs and (2) quantifying the noise input data, is crucial.

Along with SLR, other tasks within computer vision could be exploited by RNN structures. Further research on SLR is worth doing with the holistic approach. Beneficial implementations can be made with non-typical combinations of DL structures (e.g., attention mechanism, (Bi-)LSTM, (Bi-)LSTM-SVM, (Bi-)RNN) [10,54,55]. Solutions to improve quality of life have inestimable value on society, and it is expected that this work will directly impact the hearing impaired population using PSL on a daily basis in the Republic of Panama.

Author Contributions: conceptualization, A.A.T.-Q. and J.E.S.-G.; methodology, A.A.T.-Q., V.L.-C., J.C.R. and J.E.S.-G.; software, A.A.T.-Q.; validation, A.A.T.-Q., V.L.-C., J.C.R. and J.E.S.-G.; resources, A.A.T.-Q. and J.E.S.-G. writing—original draft preparation, A.A.T.-Q. and J.E.S.-G.; writing—review and editing, A.A.T.-Q., V.L.-C., J.C.R. and J.E.S.-G.; visualization, A.A.T.-Q. and J.E.S.-G.; supervision of students, J.E.S.-G. All authors have read and agreed to the published version of the manuscript.

Funding: J.E.S.-G. and J.C.R. are supported by the Sistema Nacional de Investigación (SNI) of National Secretariat for Science, Technology, and Innovation (SENACYT).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Written informed consent has been obtained from the patient(s) to publish this paper.

Data Availability Statement: The PSL images collected has been shared as a data set in the following link https://data.mendeley.com/datasets/3d4wggwh5g/1 (accessed on 20 February 2024).

Acknowledgments: The authors acknowledge administrative support provided by the Universidad Tecnológica de Panamá.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Fatmi, R.; Rashad, S.; Integlia, R. Comparing ANN, SVM, and HMM based Machine Learning Methods for American Sign Language Recognition using Wearable Motion Sensors. In Proceedings of the 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 7–9 January 2019; pp. 0290–0297. [CrossRef]
2. Sharma, S.; Kumar, K. ASL-3DCNN: American sign language recognition technique using 3-D convolutional neural networks. *Multimed. Tools Appl.* **2021**, *80*, 26319–26331. [CrossRef]
3. Rahman, M.M.; Islam, M.S.; Rahman, M.H.; Sassi, R.; Rivolta, M.W.; Aktaruzzaman, M. A New Benchmark on American Sign Language Recognition using Convolutional Neural Network. In Proceedings of the 2019 International Conference on Sustainable Technologies for Industry 4.0 (STI), Dhaka, Bangladesh, 24–25 December 2019; pp. 1–6. [CrossRef]
4. Jing, L.; Vahdani, E.; Huenerfauth, M.; Tian, Y. Recognizing American Sign Language Manual Signs from RGB-D Videos. *arXiv* , **2019**, arXiv:1906.02851.
5. Kannoth, A.; Yang, C.; Guanipa Larice, M.A. Hand Gesture Recognition Using CNN & Publication of World's Largest ASL Database. In Proceedings of the 2021 IEEE Symposium on Computers and Communications (ISCC), Athens, Greece, 5–8 September 2021; pp. 1–6. [CrossRef]
6. Perdana, I.P.; Putra, I.K.G.D.; Dharmaadi, I.P.A. Classification of Sign Language Numbers Using the CNN Method. *JITTER J. Ilm. Teknol. Dan Komput.* **2021**, *2*, 485–493. [CrossRef]
7. Marjusalinah, A.D.; Samsuryadi, S.; Buchari, M.A. Classification of finger spelling American sign language using convolutional neural network. *Comput. Eng. Appl. J.* **2021**, *10*, 93–103. [CrossRef]
8. Ariesta, M.C.; Wiryana, F.; Suharjito; Zahra, A. Sentence level Indonesian sign language recognition using 3D convolutional neural network and bidirectional recurrent neural network. In Proceedings of the 2018 Indonesian Association for Pattern Recognition International Conference (INAPR), Jakarta, Indonesia, 7–8 September 2018; pp. 16–22.
9. Ibrahim, N.B.; Zayed, H.H.; Selim, M.M. Advances, challenges and opportunities in continuous sign language recognition. *J. Eng. Appl. Sci.* **2020**, *15*, 1205–1227. [CrossRef]
10. Rastgoo, R.; Kiani, K.; Escalera, S. Sign language recognition: A deep survey. *Expert Syst. Appl.* **2021**, *164*, 113794. [CrossRef]
11. Basiri, M.E.; Nemati, S.; Abdar, M.; Cambria, E.; Acharya, U.R. ABCDM: An attention-based bidirectional CNN-RNN deep model for sentiment analysis. *Future Gener. Comput. Syst.* **2021**, *115*, 279–294. [CrossRef]
12. Yu, W.; Kim, I.Y.; Mechefske, C. Analysis of different RNN autoencoder variants for time series classification and machine prognostics. *Mech. Syst. Signal Process.* **2021**, *149*, 107322. [CrossRef]
13. Metaxas, D.; Dilsizian, M.; Neidle, C. Scalable ASL Sign Recognition using Model-based Machine Learning and Linguistically Annotated Corpora. In *8th Workshop on the Representation & Processing of Sign Languages: Involving the Language Community, Language Resources and Evaluation Conference 2018*; European Language Resources Association (ELRA): Luxembourg, 2018.
14. Rahman, M.M.; Malaia, E.A.; Gurbuz, A.C.; Griffin, D.J.; Crawford, C.; Gurbuz, S.Z. Effect of Kinematics and Fluency in Adversarial Synthetic Data Generation for ASL Recognition With RF Sensors. *IEEE Trans. Aerosp. Electron. Syst.* **2022**, *58*, 2732–2745. [CrossRef]
15. Zhang, J.; Zhou, W.; Xie, C.; Pu, J.; Li, H. Chinese sign language recognition with adaptive HMM. In Proceedings of the 2016 IEEE International Conference on Multimedia and Expo (ICME), Seattle, WA, USA, 11–15 July 2016; pp. 1–6.
16. Agrawal, S.C.; Jalal, A.S.; Tripathi, R.K. A survey on manual and non-manual sign language recognition for isolated and continuous sign. *Int. J. Appl. Pattern Recognit.* **2016**, *3*, 99–134. [CrossRef]
17. Katoch, S.; Singh, V.; Tiwary, U.S. Indian Sign Language recognition system using SURF with SVM and CNN. *Array* **2022**, *14*, 100141. [CrossRef]

18. Koller, O.; Zargaran, O.; Ney, H.; Bowden, R. Deep sign: Hybrid CNN-HMM for continuous sign language recognition. In Proceedings of the British Machine Vision Conference 2016, York, UK, 19–22 September 2016.
19. Koller, O.; Zargaran, S.; Ney, H.; Bowden, R. Deep sign: Enabling robust statistical continuous sign language recognition via hybrid CNN-HMMs. *Int. J. Comput. Vis.* **2018**, *126*, 1311–1325. [CrossRef]
20. Buttar, A.M.; Ahmad, U.; Gumaei, A.H.; Assiri, A.; Akbar, M.A.; Alkhamees, B.F. Deep Learning in Sign Language Recognition: A Hybrid Approach for the Recognition of Static and Dynamic Signs. *Mathematics* **2023**, *11*, 3729. [CrossRef]
21. Elsayed, E.K.; Fathy, D.R. Sign language semantic translation system using ontology and deep learning. *Int. J. Adv. Comput. Sci. Appl.* **2020**, *11*, 141–147. [CrossRef]
22. Abedin, T.; Prottoy, K.S.S.; Moshruba, A.; Hakim, S.B. Bangla sign language recognition using concatenated BdSL network. *arXiv* **2021**, arXiv:2107.11818
23. Fink, J.; Frénay, B.; Meurant, L.; Cleve, A. LSFB-CONT and LSFB-ISOL: Two New Datasets for Vision-Based Sign Language Recognition. In Proceedings of the 2021 International Joint Conference on Neural Networks (IJCNN), Shenzhen, China, 18–22 July 2021; pp. 1–8.
24. Calado, A.; Errico, V.; Saggio, G. Toward the Minimum Number of Wearables to Recognize Signer-Independent Italian Sign Language With Machine-Learning Algorithms. *IEEE Trans. Instrum. Meas.* **2021**, *70*, 1–9. [CrossRef]
25. Wei, F.; Chen, Y. Improving continuous sign language recognition with cross-lingual signs. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 1–6 October 2023; p. 23612.
26. Yin, A.; Zhao, Z.; Jin, W.; Zhang, M.; Zeng, X.; He, X. Mlslt: Towards multilingual sign language translation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 5109–5119.
27. Tornay, S.; Razavi, M.; Doss, M.M. Towards multilingual sign language recognition. In Proceedings of the ICASSP 2020–2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Barcelona, Spain, 4–8 May 2020; pp. 6309–6313.
28. Ge, L.; Liang, H.; Yuan, J.; Thalmann, D. Real-time 3D hand pose estimation with 3D convolutional neural networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *41*, 956–970. [CrossRef] [PubMed]
29. Zhu, Y.; Lu, W.; Gan, W.; Hou, W. A contactless method to measure real-time finger motion using depth-based pose estimation. *Comput. Biol. Med.* **2021**, *131*, 104282. [CrossRef]
30. Huang, J.; Zhou, W.; Zhang, Q.; Li, H.; Li, W. Video-based sign language recognition without temporal segmentation. In Proceedings of the AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018; Volume 32.
31. Pimentel, D.; Walker, R.; Fajardo, M. *Lengua de Señas Panameñas*; Editora Panamá América: Panama, Panama, 2018.
32. Pimentel Araúz, M.I. Sitio Web Para el Aprendizaje de Lengua de Señas Panameñas. Ph.D. Thesis, Universidad de Panamá, Vicerrectoría de Investigación y Postgrado, Panama City, Panama, 2018.
33. Flores, A.; González, E.; Pan, J.Z.; Villarreal, V.; Muñoz, L. Sistema de aprendizaje de Lengua de Señas Panameña (LSP) a través de un brazo robótico articulado con reconocimiento de gestos. In Proceedings of the Memorias de Congresos UTP, Pereira, Colombia, 11–13 September 2019; pp. 168–173.
34. Gestión Empresarial 3000. IPHE Inclusivo, 2018. Mobile App. Available online: https://play.google.com/store/apps/details?id=ca.costari.apps.ipheinclusivo&pli=1 (accessed on 15 September 2023).
35. Rodríguez-Fuentes, A.; Alaín-Botaccio, L.; García-García, F. Presentation and evaluation of a digital tool for sign language (Presentación y evaluación de una herramienta digital para la lengua de signos). *Cult. Educ.* **2022**, *34*, 658–688. [CrossRef]
36. Fuentes, A.R.; Alain, L.; García, F.G. EnSenias: Technological tool to learn, teach, improve and use Panamanian Sign Language. *Íkala* **2020**, *25*, 663–678.
37. Alaín Botacio, L. Desarrollo y Validación de una Aplicación web y cd Educativa Inclusiva Para el Aprendizaje de la Lengua de señas Panameña. Ph.D. Thesis, Universidad de Granada, Granada, Spain, 2019.
38. Bodmer, R.; Liu, L.; Liu, W.; Rangel, J.C. Sign language recognition with machine learning for elementary school children. *Rev. Iniciación Científica Edición Espec.* **2020**, *6* . [CrossRef]
39. Teran-Quezada, A.; Lopez-Cabrera, V.; Rangel, J.C.; Sanchez-Galan, J.E. Hand Gesture Recognition with ConvNets for School-Aged Children to Learn Basic Arithmetic Operations. In Proceedings of the 2022 IEEE 40th Central America and Panama Convention (CONCAPAN), Panama City, Panama, 9–12 November 2022, pp. 1–6.
40. Bazarevsky, V.; Grishchenko, I. MediaPipe Holistic—Simultaneous Face, Hand and Pose Prediction, on Device. 2020. Available online: https://ai.googleblog.com/2020/12/mediapipe-holistic-simultaneous-face.html (accessed on 15 September 2023).
41. Bazarevsky, V.; Zhang, F. On-Device, Real-Time Hand Tracking with MediaPipe. *arXiv* **2019**, arXiv:2006.10214.
42. Ablavatski, A.; Grishchenko, I. Real-Time AR Self-Expression with Machine Learning. 2019. Available online: https://blog.research.google/2019/03/real-time-ar-self-expression-with.html (accessed on 20 February 2024).
43. Bazarevsky, V.; Grishchenko, I. On-Device, Real-Time Body Pose Tracking with MediaPipe BlazePose. 2020. Available online: https://blog.research.google/2020/08/on-device-real-time-body-pose-tracking.html (accessed on 15 September 2023).
44. Zhang, F.; Bazarevsky, V.; Vakunov, A.; Tkachenka, A.; Sung, G.; Chang, C.L.; Grundmann, M. MediaPipe Hands: On-device Real-time Hand Tracking. *arXiv* **2020**, arXiv.2006.10214.
45. Bazarevsky, V.; Kartynnik, Y.; Vakunov, A.; Raveendran, K.; Grundmann, M. BlazeFace: Sub-Millisecond Neural Face Detection on Mobile GPUs. *arXiv* **2019**, arXiv:1907.05047. [CrossRef]
46. Bazarevsky, V.; Grishchenko, I.; Raveendran, K.; Zhu, T.; Zhang, F.; Grundmann, M. BlazePose: On-device Real-time Body Pose tracking. *arXiv* **2006**, arXiv:2006.10204.

47. Teran-Quezada, A.A.; Lopez-Cabrera, V.; Rangel, J.C.; Sanchez-Galan, J.E. A Collection of Basic Greetings in Panamanian Sign Language (PSL). Mendeley Data, V1. 2024. Available online: https://data.mendeley.com/datasets/3d4wggwh5g/1 (accessed on 20 February 2024). (In Spanish)
48. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
49. Simao, M.; Neto, P.; Gibaru, O. Improving novelty detection with generative adversarial networks on hand gesture data. *Neurocomputing* **2019**, *358*, 437–445. [CrossRef]
50. Shen, J.; Dudley, J.; Kristensson, P.O. The imaginative generative adversarial network: Automatic data augmentation for dynamic skeleton-based hand gesture and human action recognition. In Proceedings of the 2021 16th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2021), Jodhpur, India, 15–18 December 2021; pp. 1–8.
51. Liu, Y.; De Nadai, M.; Zen, G.; Sebe, N.; Lepri, B. Gesture-to-gesture translation in the wild via category-independent conditional maps. In Proceedings of the 27th ACM International Conference on Multimedia, Nice, France, 21–25 October 2019; pp. 1916–1924.
52. Núñez-Marcos, A.; Perez-de Viñaspre, O.; Labaka, G. A survey on Sign Language machine translation. *Expert Syst. Appl.* **2022**, *213*, 118993. [CrossRef]
53. Zhang, Y.; Zhang, Z.; Zhang, Y.; Bao, J.; Zhang, Y.; Deng, H. Human activity recognition based on motion sensor using u-net. *IEEE Access* **2019**, *7*, 75213–75226. [CrossRef]
54. Al-Qurishi, M.; Khalid, T.; Souissi, R. Deep learning for sign language recognition: Current techniques, benchmarks, and open issues. *IEEE Access* **2021**, *9*, 126917–126951. [CrossRef]
55. Nogales, R.E.; Benalcázar, M.E. Hand Gesture Recognition Using Automatic Feature Extraction and Deep Learning Algorithms with Memory. *Big Data Cogn. Comput.* **2023**, *7*, 102. [CrossRef]