Automatic Recognition of Continuous Sign Language for Public Services

Robson Silva de Souza
Dept. of Computer Engineering and Industrial Automation
School of Electrical and Computer Engineering
Campinas, SP, Brazil
robsonnddesouza@gmail.com

José Mario De Martino
Dept. of Computer Engineering and Industrial Automation
School of Electrical and Computer Engineering
Campinas, SP, Brazil
martino@unicamp.br

Janice Gonçalves Temoteo Marques
Dept. of Human Development and Rehabilitation
Faculty of Medical Sciences, University of Campinas
Campinas, SP, Brazil
janicetm@unicamp.br

Ivani Rodrigues Silva
Dept. of Human Development and Rehabilitation
Faculty of Medical Sciences, University of Campinas
Campinas, SP, Brazil
ivanirs@unicamp.br

***Abstract*—In this article, we present an automatic image recognition approach for assisting the communication between deaf people and hearing physicians. The aim of the approach is to help the interaction and exchange of information during medical interviews and in different public services, such as police departments, hospitals, and citizen service centers. Its scope is the automatic recognition of the continuous signing through the analysis of traditional video and depth data (RGB-D data). Recognition is performed by a cascade of two neural networks. First, a convolutional neural network encodes the visual input and extracts relevant features. Second, a recurrent neural network learns the mapping of the extracted features and transforms them into words. We use the Connectionist Temporal Classification approach to train the recurrent network with videos of different lengths and word sequences. Experiments on two continuous sign language datasets show the effectiveness of our approach, achieving an accuracy of around 91% in the Brazilian Sign Language (Libras) dataset and 94% in Greek Sign Language (GSL) in signer-independent continuous sign language setup.**

***Index Terms*—Brazilian Sign Language; Sign language recognition; Continuous signing; long short term memory; connectionist temporal classification**

I. Introduction

This paper builds upon and extends our previous work [1], where we present our approach, experiments, and results considering a Brazilian Sign Language (Libras) dataset in the context of anamnesis. The current paper extends the original one and also presents experiments and results, taking into account a Greek Sign Language (GSL) dataset of the public service interaction domain.

Anamnesis and clinical examination are the standard procedures of physicians to diagnose diseases and health problems of their patients. Anamnesis is a process of interviewing the patient to collect information about his/her current health complaints and medical history. The precise disclosure, correct understanding, and assessment of this information are preconditions for an effective diagnosis and the identification of the appropriate therapy. However, the effectiveness of the medical interview is jeopardized if the physician and the patient do not have a common language for communication. That is usually the case when we consider a deaf patient who has sign language as his/her first language and does not master the written language of the physician who, by his/her side, does not understand sign language. A common solution to overcome this problem is to have a sign language interpreter assisting the deaf patient during the interview. Besides the operational difficulties of organizing an interpreter, another important drawback is the uncomfortable situation created by the introduction of a third party in the medical interview. During a medical interview, the patient should feel comfortable enough to share very personal and sensitive information, providing any and all relevant information to help the doctor make a correct diagnosis. A solution to overcome this potential breach of patient-doctor confidentiality is to provide a robust computer-based solution to support the communication between physicians and deaf patients. Although the interaction between doctor and patient is a two-way process, in this article, we focus only on the issue of automatic recognition of continuous signing based on computer-based recognition of video imagery. Our main focus of interest is the continuous signing recognition of the Brazilian Sign Language. However, to provide evidence that our approach can be successfully applied to other sign languages, we also evaluated our approach using a Greek Sign Language dataset available publicly [2].

Sign languages convey information by the movement of the hands, body, and face. They are perceived by vision. There is not a single, universal sign language used worldwide by deaf people. Each country has its own sign language [3]. The sign language of a country is independent of its oral language. For example, Deaf Americans speak the American Sign Language (ASL), the Deaf in the UK use the British Sign Language (BSL), and Deaf Australians speak the Australian Sign Language (Auslan). Deaf Brazilians use the Brazilian Sign Language (Libras).

There has been increasing research interest in automatic sign language recognition in recent years. Automatic sign language recognition applies computer vision combined with machine learning techniques to analyze and translate, into a written form, videos with sign language content.

The development of robust automatic sign language recognition systems is challenging. Several techniques have been proposed for automatic sign language recognition for a variety of sign languages, including the Brazilian Sign Language (Libras). Most efforts, however, have been limited to the study of isolated sign recognition, postures representative of cardinal numbers (0 to 10), and the manual alphabet or fingerspelling. Research on continuous signing recognition is still rare.

Concerning the representation of the input data, early works in automatic recognition of sign language commonly were based on hand-crafted features that are designed beforehand by human experts to extract a given set of chosen characteristics [4]. These features were used with architectures such as Hidden Models of Markov (HMM) [5] or Conditional Random Field (CRF) [6] for sequential modeling.

In the last years, deep learning techniques have made significant advances in computational vision due to the huge increase in computational power using graphical process units (GPUs), which have added to the availability of datasets with millions of images [7], [8], [9]. From this perspective, research on continuous sign language recognition also benefited, especially concerning the alignment of frame sequences to word sequences, be it acting on systems using depth networks solely [10] or combined with HMMs [11].

The challenge faced is a somewhat overlooked problem in which the sequences of glosses generally are available but not their time limits in the videos. In order to solve this, recent models based on Recurring Neural Network (RNN) with Connectionist Temporal Classification (CTC) [12] have reached the state of the art in this task [10], [13].

Even with all the advances, work in the area still has limitations, such as the recognition system processing only cropped sequences of the hand. However, a robust system must also take non-manual expressions into account, which are fundamental components of all sign languages [14]. Another limitation is to provide extra information about the signs to the recognition system, such as medium lengths of the signs on the videos or the developments of subsystems for particular parts of the body.

This article presents a method for automatic continuous sign language recognition of Libras during medical interviews. In addition, we also verified the generalization of our method on a Greek Sign Language (GSL) dataset [2]. Applying the method, we implement an approach based on Deep Learning that is capable of finding and using extracted data from signing from full-frame sequences. Therefore, it aligns sequences of video frames displaying continuous sign language content to sequence glosses. A gloss is a word (or a couple of words) of a written language that is consistently used to label a sign within the corpus, regardless of the meaning of that sign in a particular context or whether it has been systematically modified in some way [15]. As pointed out in [16], glosses are a convenient way to write down the meaning of a sign, as they use written language to represent the signs.

The main contributions of this article are:

- The construction of a robust and representative dataset, composed of RGB information and depth of signage in Libras in order to contribute to the advancement of the research in this area.
- Execution of a Depth-Wise Separable Convolutional Network (DWSCN) based architecture, as feature extractor preprocessor. Insofar as we know, we are the first to employ this type of architecture in continuous sign language recognition systems.
- The development of a new architecture of sequential learning, based on Recurrent Neural Networks and Connectionist Temporal Classification, which learn to find and store relevant data in its memory cells from the full-frame sequences, without importing in its subsystems structures that process image patches.

The remainder of the paper is organized as follows: Section II contains a review of relevant related work. Section III presents our approach. Section IV describes the experiments performed, and Section V presents the conclusions.

II. Related Work

The recognition of continuous signing is a far more complex task than the recognition of isolated signs, requiring more sophisticated methods to deal with the dynamics of production and the transition between signs. As previously stated, continuous signing recognition systems are more appropriate for real-world scenarios of interpersonal communication. However, it is observed that there is still little research that seeks to solve this problem. In the following paragraphs, we present approaches aimed at recognizing continuous signing based on computer vision.

Until recently, research was based on hand-crafted features for spatio-temporal representations in combination with sequence modeling methods. Among these, we can mention the algorithms based on threshold models such as [17]. Yang and colleagues focus on parameters of threshold models for labels of epenthetic movements, in which they perform tests using HMM, calculating the similarity between the sign model and the test sequence [17]. Techniques based on dynamic time distortion (DTW) were also widely used, which measure the similarity between two sequences of temporal data based on the minimum distance between them; therefore, the data have their lengths altered in order to obtain the best mapping. Among these works are those of Zhang and colleagues [18], that use DTW for the recognition of Chinese Sign Language (CSL) sentences using the Kinect. In [19], an HMM is used to learn hand and elbow features. The authors propose thresholds that describe the probability of removing transitional motion from a given video segment and use DTW to determine the endpoint for each candidate sign. The final recognition is obtained by concatenating the most possible signs.

Research using deep learning models has increased considerably in recent years, whether acting in independent systems or combined with HMMs in the then called hybrid approach. Using a German Sign Language recognition dataset called RWTH-PHOENIX-Weather, Koller and colleagues [20], [21], [22], [23] built a hybrid architecture, consisting of a convolutional neural network (CNN) to learn representations of frame-by-frame labels of hand-cut sequences and HMMs to model time dependencies. They trained a network with frame sequences from a label assignment initialization called flat-start. In [24] they use state alignment per frame, provided by an HMM as frame labeling to train the neural networks.

Although HMMs have achieved good results in several tasks that involve sign language, [25] indicates that traditional approaches to Markov models are limited because their states must be designed from a modest-sized state space, and that the dynamic program algorithm used to perform efficient inference with HMMs scales in time with quadratic time complexity $O(S^2)$ [26]. Moreover, Graves and colleagues claim that these models require assumptions that observations in HMM are independent to make inference treatable [12].

On the other hand, RNNs show great capacity for sequential learning. According to [12], they are not like hybrid systems, which inherit the previously mentioned inconveniences of HMMs. Furthermore, hybrid RNN-HMM systems are not able to exploit the full potential of RNNs for sequence modeling.

Another promising model is the CTC Network, originally proposed for speech recognition [12]. CTC is an ideal method for tasks where data is poorly labeled, i.e., it does not require *a priori* alignments between input and output sequences and allows recurring networks to be trained with different video lengths and label sequences.

Camgoz and colleagues propose an approach that breaks down the problem of recognizing signs into a series of expert systems called subunits [27]. Each subunit consists of three layers of neural networks: a Convolutional Neural Network (CNN) for extraction of spatial features; a Bidirectional Long Short-Term Memory (BLSTM) [28], an extension of LSTM [29] that temporarily models the features; and a loss layer based on the CTC. A recent work, [11], also uses CNN and LSTM but encapsulated it in an HMM model following the hybrid approach used in his previous work, this time exploring sequential parallelism to learn sign language, mouth shapes, and hand shape classifiers.

The works [10], [30]–[33] use CNNs as feature extractors, a 3D CNN model, or a 3D residual convolutional network (3D-ResNet). For modeling and sequential learning, they use Dilated Convolutional Networks or RNNs such as LSTM, Gated Recurrent Unit (GRU) [34] and their variants in combination with the CTC algorithm. Among these approaches, [13] is the one that achieved the best performance in the RWTH-PHOENIX-Weather dataset and also in a set of images captured by the Kinect called CSL-25K, which covers 100 daily life sentences expressed in Chinese Sign Language (CSL).

In our proposal, we also use recurrent neural networks with CTC, but differently from the other approaches, we apply a depth-wise separable convolutional network that contains far fewer parameters and is computationally cheaper than the state-of-the-art convolutional neural networks, as for example, VGG16 [35], ResNet50 [36], and InceptionV3 [37].

III. Method

In this section, we present the guidelines for the construction of a dataset and our approach for recognizing continuous signing. The approach includes a CNN-based model for features extraction and an RNN architecture for learning the spatial-temporal dependencies that exist between the sentence signs. To solve the alignment problem between the probability sequences in the RNN outputs with the sequences of glosses, we used CTC.

A. Dataset construction

A dataset composed of Libras sentences related to medical interviews is fundamental for developing and testing our approach. No publicly available image databases of continuous signing in Libras have been found.

Through the study of existing datasets of other sign languages [38], [39] and with the intent of meeting our objectives, we developed specifications to be followed for the construction of our dataset. The proposal is to develop a robust dataset that simulates the internal environment of a clinic with artificial lighting, in which the deaf volunteer or interpreter performs the sign naturally.

Fig. 1 shows the execution flow. Thereafter, each module will be described in detail.

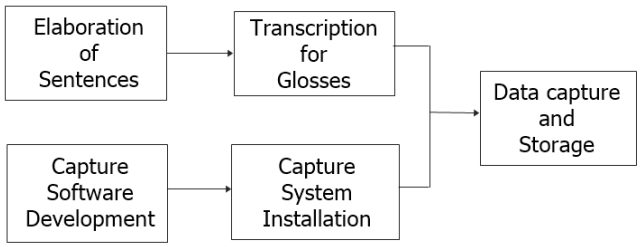


Fig. 1. Execution flow for the construction of the dataset

Sentences elaboration. Comprised of the elaboration of sentences in the Portuguese language from the answers of a patient in the context of a medical consultation (general practitioner). The sentences are established through the study of signs and manifested individual symptoms accordingly to the anamnesis medical procedure described in [40] and [41]. Following this procedure, our protocol encompasses:

- Main complaint: Brief phrase from the patient that explains his reason for looking for the physician;
- History of current sickness: Description of the main complaint concerning the chronology, when symptoms first manifested;

Transcription of the sentences in Portuguese to glosses. Glosses created through the assistance of a fluent sign lan-

guage specialist. The right columns on Tables I and II present the transcriptions of the sentences from the previous stage.

TABLE I
EXAMPLES OF SENTENCES DEVISED IN PORTUGUESE LANGUAGE AND ITS TRANSCRIPTIONS TO GLOSSES.

#	Target	Prediction
1	Eu tenho febre	EU FEBRE
2	Eu estou fraco	EU FRACO
3	Eu estou com diarréia	EU TER DIARRÉIA
4	Eu tenho manchas no rosto	EU TER MANCHA-ROSTO
5	Eu estou com tosse	EU TOSSE
6	Meu braço direito dói	MEU BRAÇO-DIREITO DOR
7	Meu braço esquerdo dói	MEU BRAÇO-ESQUERDO DOR
8	Meu dente dói	MEU DENTE DOR
9	Meu olho direito está vermelho	OLHO-DIREITO APONTAR VERMELHO TER
10	Meu olho esquerdo está vermelho	OLHO-ESQUERDO APONTAR VERMELHO TER
11	Minha urina está marrom	MEU XIXI COR MARROM
12	Minha boca está sangrando	MINHA BOCA SANGUE TER
13	Começou a um dia	COMEÇAR UM-DIA PASSADO
14	Começou há uma hora	COMEÇAR UMA-HORA PASSADO
15	Começou agora	COMEÇAR AGORA
16	Começou anteontem	COMEÇAR ANTEONTEM
17	Começou no domingo passado	COMEÇAR DOMINGO PASSADO
18	Faz um ano	COMEÇAR JÁ TER TEMPO UM ANO
19	Faz um mês	COMEÇAR JÁ TER TEMPO UM MÊS
20	Começou na quarta-feira passada	COMEÇAR QUARTA-FEIRA PASSADO

TABLE II
EXAMPLES OF SENTENCES DEVISED IN PORTUGUESE LANGUAGE AND ITS TRANSCRIPTIONS TO GLOSSES - VERSION IN ENGLISH

#	Target	Prediction
1	I have fever	ME FEVER
2	I am weak	ME WEAK
3	I have diarrhea	ME HAVE DIARRHEA
4	I have spots on the face	ME HAVE SPOT-FACE
5	I have spots on the face	ME COUGH
6	My right arm hurts	MY RIGHT-ARM PAIN
7	My left arm hurts	MY LEFT-ARM PAIN
8	My tooth hurts	MY TOOTH PAIN
9	My right eye is red	RIGHT-EYE POINT RED HAVE
10	My left eye is red	LEFT-EYE POINT RED HAVE
11	My urine is brown	MY PEE BROWN COLOR
12	My mouth is bleeding	MY MOUTH BLOOD HAVE
13	It started a day ago	START ONE-DAY PAST
14	It started an hour ago	START ONE-HOUR PAST
15	It started now	START NOW
16	It started the day before yesterday	START THE-DAY-BEFORE-YESTERDAY
17	It started last Sunday	START SUNDAY LAST
18	It is been a year	START ALREADY HAVE TIME ONE-YEAR
19	It is been a month	START ALREADY HAVE TIME ONE-MONTH
20	It started last Wednesday	START WEDNESDAY LAST

Capture device and development of the capture software. The data recording is made through the Kinect device v2 for Windows. The capture application is developed using Kinect's own software development kit (SDK). This application captures and stores RGB images, depth images, and mapped images (RGB images mapped on the depth images), in which all pixels not belonging to the signer are converted to black.

Installation of the capture system. The Libras signing recordings executed by the volunteer are made in a laboratory, with artificial illuminations and homogeneous scene background.

The Kinect is fixed on an adjustable photographic tripod and positioned at approximately 1,2m high and to a distance of around 1,3m from the signer, as seen in figure 2. These positions are determined taking in consideration the capture hardware characteristics as, i.e., minimum distance (0,5m) and maximum depth (4,5m), horizontal (70 degrees), and vertical (60 degrees) field of view.

The Kinect is connected to a computer with USB 3.0 port, with 64-bit (x64) operational system, physical dual-core 3.1 GHz processor, 4GB random access memory (RAM) or more, and a graphics adapter with DirectX 11 support [42].

In addition, the environment also has a conventional monitor (2D), in which the sentences to be signed are exhibited to the volunteer during the recordings.

Capture and data storage. The only request to the volunteer is to wear plain shirts of any color other than black, as some black shirt dyes can absorb infrared light, impairing Kinect's capacity of tracking the user (depth data) [42]. A Libras interpreter teacher member of the research team helps with the video acquisition. The position where the volunteer must be positioned during the recordings is indicated by ground marks. This position, called rest position, consists of standing in front of the capture device with lowered hands, as shown in Figure 2.

Fig. 2. Positioning and relative distances of the signer to the device.

Once the volunteer is properly positioned, the sentences are exhibited on the monitor to the volunteer. After each exhibited sentence, the volunteer must sign it naturally and return to the rest position after signing it. During the signing, the images are captured and stored on the computer. The participant is asked to remain in the rest position for a few seconds so the capture system can finish the data storage. The next sentence is then exhibited on the monitor, and the recording procedure as described above is repeated.

Figure 3 illustrates the process of capture and data storage.

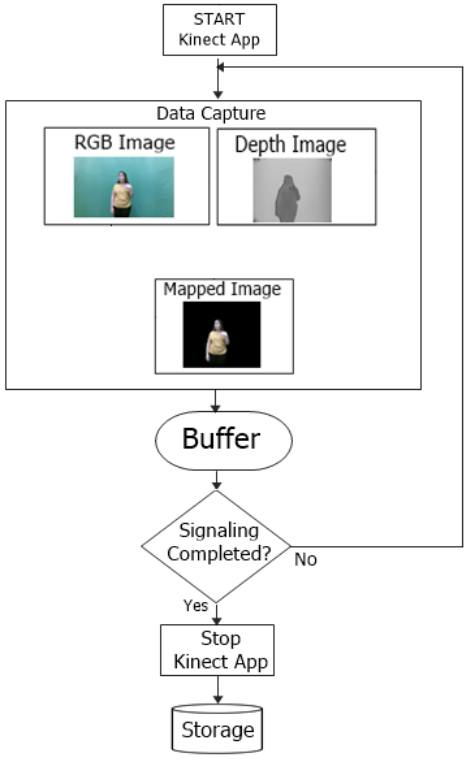


Fig. 3. Process of capture and data storage.

B. Our approach

The approach that recognizes continuous sign language includes a CNN-based model for features extraction and an RNN architecture for learning the spatial-temporal dependencies that exist between the sentence signs. To solve the alignment problem between the probability sequences in the RNN outputs with the sequences of glosses, we used CTC.

Fig. 4 presents a general view of our approach composed of three main models. The first comprises spatial modeling, while the others encompass sequential learning and a CTC loss layer to decode categorical probabilities in sequences of glosses.

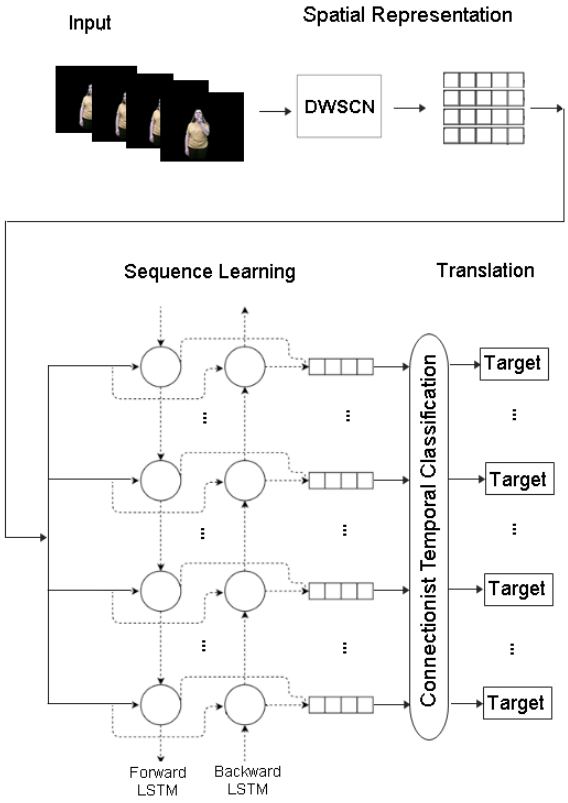


Fig. 4. Overview of our continuous sign language recognition approach

Features extraction. DWSCN is used for representations of spatial features of the frame sequences. The pre-trained MobileNetV1 [43] operational model is among the models based on the DWSCN. The use of pre-trained models enables the development of efficient models in situations of limited data availability, in addition to reducing processing time [44].

MobileNetV1 was pre-trained on ImageNet [7]. MobileNetV1 has a reduced size (17MB) and reduced number of parameters (4,2 million) when compared to other state-of-the-art models. Figure 5 shows the components of this architecture, in which each convolutional layer is followed by Batch Normalization and ReLU activation function.

MobileNetV1 expects color input images with size of 224 x 224 x 3. Thus, the obtained images with Kinect must be rescaled to this size.

The pixel values are scaled between 0 and 1, and then each channel is normalized with respect to the ImageNet dataset according to (1), (2) and (3), where p_r, p_g and p_b are the

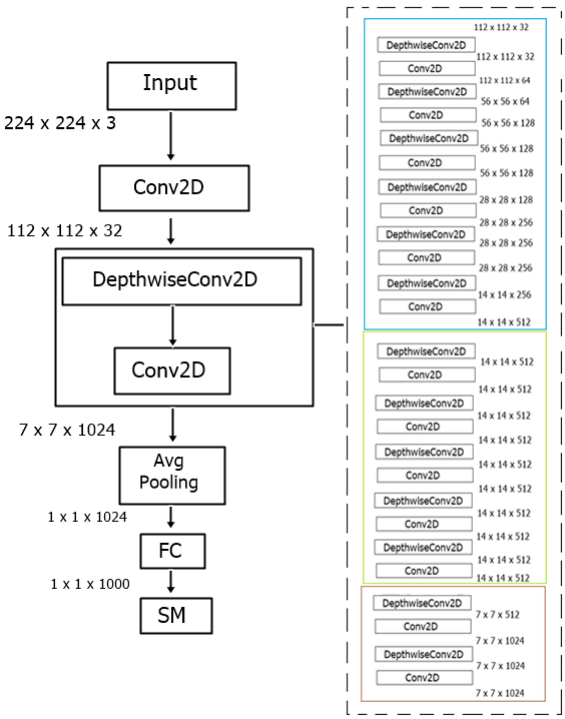


Fig. 5. MobileNetV1 Architecture

pixel values for the red, green and blue channels of the dataset images.

$$band_r = \frac{p_r/255 - mean_r}{std_r} \quad (1)$$

$$band_g = \frac{p_g/255 - mean_g}{std_g} \quad (2)$$

$$band_b = \frac{p_b/255 - mean_b}{std_b} \quad (3)$$

The averages $mean_r$, $mean_g$ and $mean_b$ with respect to the Imagenet are 0.485, 0.456, 0.406, respectively. The standard deviations std_r, std_g and std_b are equal to 0.229, 0.224, 0.225.

To use MobileNet as a feature extractor preprocessor, the softmax classification layer (SM) and the completely connected layer (FC) have been removed, keeping all the depthwise separable convolution blocks and the Average Pooling layer.

All dataset images are processed by the resulting model. As the last layer has 1024 nodes, each image will be represented as a 1024 value vector. Each video sample results in a three-dimensional array of dimensions equal to 1 x number of frames x 1024 features. Since the numbers of frames are different between the videos, the padding in each array has been performed to allow the concatenating of all feature arrays.

All arrays are then concatenated and stored for a single NumPy array [45] in the standard binary file format (NPY).

The glosses are coded in categorical variables and, together with the feature arrays, are used as input to train our model based on recurring neural networks. This is an overlooked learning problem as the gloss sequences are available but not its time limits.

Sequential learning. Our approach uses BLSTM to model the correspondences between the input sequences and output glosses. This architecture is capable of storing data for long periods of time and try to avoid the explosion of the gradient, a common problem of the Vanilla Neural Networks.

To implement a BLSTM network, it takes two parallel layers of LSTM cells, backward LSTM and forward LSTM, each of them being responsible for processing the information in the direction of time. The final hidden layer is given by the concatenation of the two networks.

The memory neurons of an LSTM are called cells. Fig. 6 presents the structure of a BLSTM network and highlights one single memory cell. The cells are capable of storing data in the course of a sequence through units called gates. According to [46], these units calculate the weights that connect them to avoid the gradient degradation through parameterized or manually chosen values.

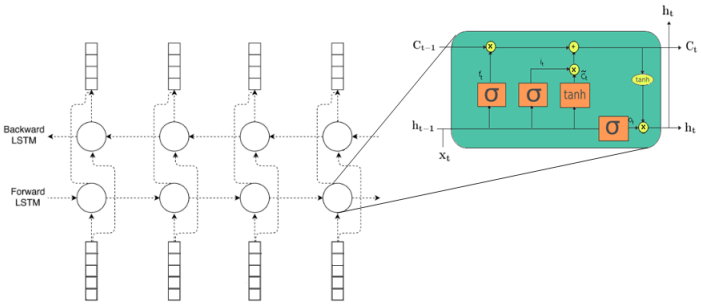


Fig. 6. BLSTM network structure, highlight to a single memory cell

The memory cell gates are composed of a sigmoid activation function and a multiplication operation between the weights and the inputs given by the Hadamard product. The operations that happen inside of an LSTM cell is detailed thereafter: The forget gate given by (4) decides which elements from the memory cell of previous state, C_{t-1}, are discarded.

$$f_t = \sigma(W_f.[h_{t-1}, x_t] + b_f) \quad (4)$$

The input gate given by (5) selects which information is going to be stored, multiplying its result by the candidate of the current memory cell, given by (6).

$$i_t = \sigma(W_i.[h_{t-1}, x_t] + b_i) \quad (5)$$

$$\widetilde{C}_t = \tanh(W_c.[h_{t-1}, x_t] + b_c) \quad (6)$$

The hidden state of the current memory cell is given by (7), which combines the previous operations, that is, the process of forgetting $f_t * C_{t-1}$ and the process of insertion of new information on memory cell, $i_t * \widetilde{C}_t$.

$$C_t = f_t * C_{t-1} + i_t * \widetilde{C}_t \quad (7)$$

After these procedures, the hidden state of the current cell is built, h_t, given by (9), multiplying the output gate, (8), by the value of the hidden state of the memory cell, C_t.

$$o_t = \sigma(W_o.[h_{t-1}, x_t] + b_o) \tag{8}$$

$$h_t = o_t * \tanh(C_t) \tag{9}$$

In the equations above, W_f, W_i, W_c, W_o, b_f, b_i, b_c, b_o are the values of the network weights and bias.

A softmax activation function on a fully connected layer is used in the network output and is applied to each time frame.

Connectionist Temporal Classification. In the BLSTM training phase, CTC is used to calculate the cost value. During prediction, it decodes the probability matrices of the softmax function in gloss sequences.

To allow the CTC algorithm to decode the target sequence, one more unit is introduced to the total number of labels in the softmax output layer. This unit refers to a token named blank, that models the transitions between different labels.

Let us consider the mapping of the input frames sequence $X = [x_1, x_2, ..., x_T]$, for the sequences of output words $Y = [y_1, y_2, ..., y_T]$. The CTC cost function for a pair (X, Y) has the conditional probability $p(Y/X)$ equal to the sum of all the valid paths $A \in A_{XY}$, calculating the probability $p_t(a_t|X)$ to a single step-by-step alignment following (10).

$$p(Y/X) = \sum_{A \in A_{XY}} \prod_{a}^{b} p_t(a_t|X) \tag{10}$$

For a training set M, the model parameters are tuned to minimize the negative log-likelihood. That way, the CTC objective function is given by (11).

$$Loss_{CTC} = \sum_{(X,Y) \in M} -\log p(Y/X) \tag{11}$$

To calculate the CTC loss efficiently, the Forward-Backward algorithm given in [12] is used.

IV. Experiments

This section reports on the experiments performed and the performance of our architecture in continuous Libras signing recognition. We evaluated the performance of our method on the GSL dataset [2].

A. Datasets

In order to develop and test our approach, 280 sentences signed in Libras by a professional interpreter were captured, corresponding to 5 repetitions of 56 sentences. 42663 frames were obtained at a rate of 30 fps. The statistical details are presented in Table III.

TABLE III
Statistics of our dataset

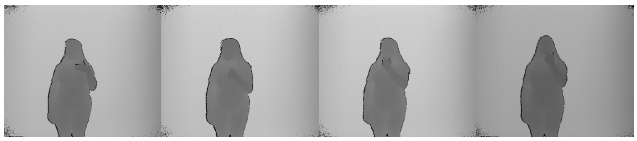
Statistics	Data
Sentences	280
Vocabulary	67
Frames	42663
Glosses per Sentence	2 - 6
Frames per Sentence	124 - 277

RGB images obtained by Kinect have a resolution of 1920 x 1080 pixels. But for better performance of our data capture and storage application, at run time, these images were rescaled to 640 x 360 pixels. Table IV summarizes the written data and its corresponding sizes.

TABLE IV
Specifications resulting from our data capture and storage application.

Data	Size
RGB Image	640 x 360
Depth Image	512 x 424
Mapped Image	512 x 424

Figure 7 presents 3 types of captured data: 7(a) Depth images, 7(b) RGB images and 7(c) Mapped images. In this paper, only mapped images are used.



(a) Depth Image

(b) RGB Image

(c) Mapped Image

Fig. 7. Examples of data captured by the Kinect device.

We also evaluated our method on the GSL, in the Signer-independent continuous sign language recognition setup [2].

The GSL dataset is publicly available, and the captured data involve cases of deaf people interacting with different public services (police departments, hospitals, and citizen service centers). Recordings are performed in the laboratory using an

Intel RealSense D435 RGB+D at a rate of 30 fps. Data is acquired at a spatial resolution of 848x480 pixels. In total, 7 signers participate in data capture. Fig. 8 illustrates examples of frames from the GSL dataset.

Fig. 8. Example frames of the GSL dataset

The statistical details are presented in Table V:

TABLE V
STATISTICS OF GSL DATASET

Statistics	Data
Sentences	10295
Vocabulary	310
Frames	1036155
Mean sentence lenght	4,23
Frames/Sentence	6 - 615

B. Evaluation Metrics

The Word Error Rate (WER) is the metric widely [30], [31], [47], [48], [33], [32], [10], [13], [27], [21], [22], [23] used in continuous sign language recognition work. We also use this metric in our evaluation. The WER is given by (12).

$$WER = \frac{I + D + S}{N} \tag{12}$$

Where I is the number of errors entered, D is the number of deletion errors, S is the number of substitution errors, and N is the total number of glosses in the reference sentence.

The accuracy is given by (13).

$$acc = 1 - WER \tag{13}$$

C. Training and Evaluation

We performed experiments on an Nvidia RTX 2080Ti, and the model is implemented in the Keras framework [49], using tensorflow [50] as a backend.

Both in the Libras dataset and in the GSL dataset the simulations performed processed the mapped images. Initially, these images were resized to 224 X 224 pixels, dimensions expected by the MobileNetV1 network, using bilinear interpolation. Then the images are converted to Numpy array, the pixel values are scaled between 0 and 1, and then each channel is normalized in relation to the ImageNet dataset.

After spatial modeling with our structure based on DWSCN, the resulting feature matrix has a dimension equal to the number of samples x time steps x features.

Libras dataset. In our experiments, we used 80 percent of the data (224 sentences) for the training set and 20 percent for the test set (56 sentences).

According to [51], training small datasets offers some challenges, as the network effectively memorizes the training dataset. The author recommends that adding noise is an approach to improve the generalization error and to enhance the structure of the mapping problem during learning. Thus, we applied Gaussian noise, at the entrance of the BLSTM network, with a standard deviation of 0.5 during the training phase.

The training of our BLSTM architecture is performed by implementing the backpropagation algorithm through time, [52]. The initialization for the recurrent weights matrix is orthogonal [53], for non-recurring weights the glorot uniform [54] and the vector bias is initialized with zeros. The optimizer used is the Root Mean Square Propagation (RMSprop) [55] with a learning rate of 0.01, a discounting factor of 0.9, a momentum of zero, (default values in the framework), and a batch size of 82. Then, we use the CTC beam decoder described in [56] to decode sentences with a beam width of 10.

For the aforementioned configurations, dozens of experiments were carried out using different network topologies, with a maximum of 4 layers (1 to 3 recurring layers and a completely connected layer) and the number of neurons equal to powers of 2 in the range of 2 to 512. The last layer is fixed with 68 neurons (one for each vocabulary label plus the blank label). Given the stochastic nature of the algorithms used, repetitions of the tests are performed in order to determine the most promising models.

In order to detect overfitting and determine the most promising models, a validation set is prepared, based on the training set, consisting of 60 sentences. During the training, at the end of each epoch, the value of the loss CTC is calculated in the validation set, and the best model in each training is determined according to the lowest value of the loss in that set.

Also, to identify and soften the effect of overfitting, we used the method of regularization called dropout, presented in [57]. Dropout values equal to 0.5 were applied for both recurrent and non-recurrent connections.

Among the best models that fit the data, the simplest model, that is, with the least hyperparameters, is considered the most plausible to be used in the test set.

GSL dataset. To verify the generalization, we also evaluate our method in the GSL dataset in the Signer-independent continuous sign language recognition setup [2]. The recordings

of a signer are separated into the validation set (588 sentences) and test set (881 sentences). The recordings of the other 6 signers comprise the training set (8821 sentences).

Due to memory constraints, we do not use all video frames of the GSL. Instead, we calculate the average of existing frames in the training set and use this value as the maximum amount for each video. As a result, all videos with more than 100 frames are reduced to 100, in such a way that the selected frames are determined from a linear spacing. Videos with less than 100 frames remain unchanged.

The random weight distribution types for the recurrent weights matrix, non-recurring weights, and the vector bias are the same as used in the Libras dataset.

Simulations were performed using the RMSProp optimizer and also with the optimizer Adaptive with Momentum (Adam) [58] and its variants Adadelta [59], Adagrad [60] and Adamax [58].

The hyperparameters of the optimizers' exponential decay rate for the 1st moment and the 2nd moment estimates were set at 0.9 and 0.999 respectively, while the learning rate varied from 0.01 to 0.00001. The batch size values in the simulations were: 128, 256, 512 and 1024. We use the CTC beam decoder to decode sentences with beam width of 100.

In the same way as we did in the Libras dataset, we used several topologies with a maximum of 5 layers (2 to 4 recurrent layers and a fully connected layer) and the number of neurons equal to a power of 2 in the range of 32 to 1024.

In order to reduce overfitting, in addition to applying the regularization dropout, we use the L2 regularization factor [61], varying its value from 0.1 to 0.0001.

D. Results

Libras dataset. Our best result was achieved by configuring two recurrent layers with 32 and 64 neurons, respectively. At the end of 30000 epochs, it was determined that the best model corresponds to epoch 21422. The values of the initial weights and the settings referring to that model were saved and stored for reproducibility, as well as for use in the unseen data set during the training.

Of the 56 sentences in the test set, 11 obtained some kind of error in the model prediction. The average WER was 8.92% and therefore, an accuracy of 91.07%. In Table VI we can observe some errors found, comparing the results of the model with the ground-truth sentences. Bold words are associated with errors in prediction. Table VII presents the equivalent results in English.

Therefore, the errors found were: 13 substitutions, 2 insertions, and no deletions. Low values in relation to the total amount of glosses existing in the dataset demonstrate the effectiveness of our architecture.

GSL dataset. Our best result is achieved by configuring two recurrent layers with 256 and 256 neurons, respectively. At the end of 40,000 epochs, it was determined that the best model corresponds to epoch 28371. This is achieved using the Adam optimizer, with a learning rate of 0.0001 and L2

TABLE VI
SENTENCES WITH PREDICTION ERRORS

#	Target	Prediction
1	COMEÇAR ANTEONTEM	COMEÇAR **ONTEM**
2	COMEÇAR QUINTA-FEIRA PASSADA	COMEÇAR **TERÇA-FEIRA** PASSADA
3	COMEÇAR SEGUNDA-FEIRA PASSADA	COMEÇAR **QUARTA-FEIRA** PASSADA
4	COMEÇAR TERÇA-FEIRA PASSADA	COMEÇAR **QUINTA-FEIRA** PASSADA
5	MAU-HÁLITO FEDOR TER	MAU-HÁLITO FEDOR **VERMELHO** TER
6	MEU DENTE DOR	MEU **COSTAS TER**
7	MEU NARIZ DOR	MEU **OLHO-ESQUERDO INCHADO**
8	OLHO-DIREITO APONTAR VERMELHO TER	OLHO-DIREITO APONTAR VERMELHO **SABOR NÃO-TER**
9	MEU OLHO-DIREITO DOR	MEU OLHO-DIREITO **INCHADO**
10	MEU OMBRO-DIREITO DOR	MEU **PESCOÇO** DOR
11	MEU PESCOÇO DOR	MEU **GARGANTA MARROM**

TABLE VII
SENTENCES WITH PREDICTION ERRORS - VERSION IN ENGLISH

#	Target	Prediction
1	START BEFORE-YESTERDAY	START **YESTERDAY**
2	START THURSDAY PAST	START **TUESDAY** PAST
3	START MONDAY PAST	START **WEDNESDAY** PAST
4	START TUESDAY PAST	START **THURSDAY** PAST
5	BAD-BREATH BAD-SMELL HAVE	BAD-BREATH BAD-SMELL **RED** HAVE
6	MY TOOTH PAIN	MY **BACK HAVE**
7	MY NOSE PAIN	MY **LEFT-EYE SWOLLEN**
8	RIGHT-EYE POINT RED HAVE	RIGHT-EYE POINT RED **FLAVOR DO-NOT-HAVE**
9	MY RIGHT-EYE PAIN	MY RIGHT-EYE **SWOLLEN**
10	MY RIGHT-SHOULDER PAIN	MY **NECK** PAIN
11	MY NECK PAIN	MY **THROAT BROWN**

regularization factor in each recurring layer with a value of 0.001.

Of the 881 sentences in the test set, 790 sentences were correctly predicted, and 91 sentences were obtained some kind of error in the model prediction. These 91 sentences total 372 glosses, of which 227 gloss predictions are correct, and the errors found were: 79 substitutions, 16 insertions, and 50 deletions. Therefore, the average WER was 6.0% and, consequently, the accuracy 94.0%.

In table VIII, we quantitatively compare our results with the best results obtained in [2]. In this work the authors implement recent deep neural network methods for continuous sign language recognition. Such methods are: SubUNets [27], GoogLeNet+TConvs [10], 3D-ResNet+BLSTM [31] and I3D+BLSTM [62]. Sequence alignment and decoding use CTC [12], Entropy Regularization CTC [63] and Stimulated CTC [64].

TABLE VIII
COMPARISON WITH METHODS ON GSL IN THE SIGNER-INDEPENDENT CONTINUOUS SIGN LANGUAGE RECOGNITION SETUP

Method	WER (%)
SubUNets+CTC	20.58
3D-ResNet+BLSTM+EnStimCTC	24.01
GoogLeNet+TConvs+EnCTC	6.75
I3D+BLSTM+EnStimCTC	6.1
DWSCN+BLSTM+CTC (Our)	6.0

Compared to the mentioned methods, our method achieves the best performance in the test set in the GSL dataset in the Signer-independent continuous sign language recognition setup. This is achieved using only the traditional CTC criterion (computationally cheaper) rather than using extensions to it. Furthermore, there is no need to pretrain the model in the respective isolated sign dataset version in our method. Another advantage is that we do not need data augmentation techniques, and we use less data than what we provide.

V. CONCLUSIONS

In this article, we presented an approach for the recognition of continuous sign language. This approach receives sequences of images of a person communicating in sign language and translates continuous signing into written language. Our approach produces state-of-the-art comparable results.

In general, when compared to other approaches in the literature, our approach demonstrates a series of advantages:

i) It does not depend on the extraction of manual features, specifically designed for a domain and laboriously calculated from the geometry of the hands and arms.

ii) It takes into account characteristics related to non-manual expressions, such as movements of the face, eyes, head, and torso, instead of using only continuous sequences of the hands.

iii) Contrary to other studies' continuous signing recognition, which performs the feature extraction process in video segments related to isolated signs, our spatial representation module is processed on the entire video. Our choice is due to the fact that video representation based on fixed-length signs can compromise the continuous recognition of signing in real situations since the same sign varies in length in a video, even when performed by the same person in different situations

iv) Our spatial modeling, which is based on depthwise separable convolutions, reduces the latency and favors the development of real-time sign recognition because of the accuracy and the number of parameters and demanded calculations. This is a great advantage when compared to other convolutional neural networks.

v) Our architecture based in BLSTM with CTC learns to find and store information relevant memory cells from the data channels included in full-frame sequences. This is done without injecting subsystems in its structure that process image patches. Consequently, our approach presents a greater capacity for temporal learning compared to studies that import extra data in its system to ease the learning.

Our approach demonstrates the potential to be applied in signing recognition on heterogeneous backgrounds due to the use of Kinect, which performs the segmentation of the individual while capturing the depth and color of images. In our upcoming work, we intend to include more signage and diversify the recording scenarios of our dataset images, as well as increase the vocabulary in order to maximize the robustness of our recognition approach.

ACKNOWLEDGMENT

This study was financed in part by the Coordination for the Improvement of Higher Education Personnel - CAPES, Brazil - Finance Code 001.

REFERENCES

[1] R. S. de Souza, J. M. De Martino, J. G. T. Marques, and I. R. Silva, "Automatic recognition of continuous signing of brazilian sign language for medical interview," in *Sixth International Conference on Informatics and Assistive Technologies for Health-Care, Medical Support and Wellbeing HEALTHINFO*, Barcelona, Spain, 2021.

[2] N. M. Adaloglou, T. Chatzis, I. Papastratis, A. Stergioulas, G. T. Papadopoulos, V. Zacharopoulou, G. Xydopoulos, K. Antzakas, D. Papazachariou, and P. none Daras, "A comprehensive study on deep learning-based methods for sign language recognition," *IEEE Transactions on Multimedia*, 2021.

[3] N. Timmermans *et al.*, *The status of sign languages in Europe*. Council of Europe, 2005.

[4] L. Nanni, S. Ghidoni, and S. Brahnam, "Handcrafted vs. non-handcrafted features for computer vision classification," *Pattern Recognition*, vol. 71, pp. 158–172, 2017.

[5] B. Bauer and H. Hienz, "Relevant features for video-based continuous sign language recognition," in *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, pp. 440–445, IEEE, 2000.

[6] J. D. Lafferty, A. McCallum, and F. C. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the Eighteenth International Conference on Machine Learning*, pp. 282–289, 2001.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, Ieee, 2009.

[8] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, pp. 1097–1105, 2012.

[10] R. Cui, H. Liu, and C. Zhang, "A deep neural framework for continuous sign language recognition by iterative training," *IEEE Transactions on Multimedia*, 2019.

[11] O. Koller, C. Camgoz, H. Ney, and R. Bowden, "Weakly supervised learning with multi-stream cnn-lstm-hmms to discover sequential parallelism in sign language videos," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.

[12] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proceedings of the 23rd International Conference on Machine learning*, pp. 369–376, ACM, 2006.

[13] H. Zhou, W. Zhou, Y. Zhou, and H. Li, "Spatial-temporal multi-cue network for continuous sign language recognition," *arXiv preprint arXiv:2002.03187*, 2020.

[14] K. Brown, *Encyclopedia of language and linguistics*, vol. 1. Elsevier, 2005.

[15] T. Johnston, "From archive to corpus: transcription and annotation in the creation of signed language corpora," in *Proceedings of the 22nd Pacific Asian Conference on Language, Information, and Computation*, pp. 16–29, 2008.

[16] A. Baker, B. van den Bogaerde, R. Pfau, and T. Schermer, *The linguistics of sign languages: An introduction*. John Benjamins Publishing Company, 2016.

[17] W. Yang, J. Tao, and Z. Ye, "Continuous sign language recognition using level building based on fast hidden markov model," *Pattern Recognition Letters*, vol. 78, pp. 28–35, 2016.

[18] J. Zhang, W. Zhou, and H. Li, "A new system for chinese sign language recognition," in *2015 IEEE China Summit and International Conference on Signal and Information Processing (ChinaSIP)*, pp. 534–538, IEEE, 2015.

[19] J. Zhang, W. Zhou, and H. Li, "A threshold-based hmm-dtw approach for continuous sign language recognition," in *Proceedings of International Conference on Internet Multimedia Computing and Service*, p. 237, ACM, 2014.

[20] O. Koller, J. Forster, and H. Ney, "Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers," *Computer Vision and Image Understanding*, vol. 141, pp. 108–125, 2015.

[21] O. Koller, R. Bowden, and H. Ney, "Automatic alignment of hamnosys subunits for continuous sign language recognition," *LREC 2016 Proceedings*, pp. 121–128, 2016.

[22] O. Koller, H. Ney, and R. Bowden, "Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3793–3802, 2016.

[23] O. Koller, O. Zargaran, H. Ney, and R. Bowden, "Deep sign: Hybrid cnn-hmm for continuous sign language recognition," in *Proceedings of the British Machine Vision Conference 2016*, 2016.

[24] O. Koller, S. Zargaran, and H. Ney, "Re-sign: Re-aligned end-to-end sequence modelling with deep recurrent cnn-hmms," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4297–4305, 2017.

[25] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," *arXiv preprint arXiv:1506.00019*, 2015.

[26] A. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Transactions on Information Theory*, vol. 13, no. 2, pp. 260–269, 1967.

[27] N. C. Camgoz, S. Hadfield, O. Koller, and R. Bowden, "Subunets: End-to-end hand shape and continuous sign language recognition," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 3075–3084, IEEE, 2017.

[28] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional lstm and other neural network architectures," *Neural networks*, vol. 18, no. 5-6, pp. 602–610, 2005.

[29] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[30] J. Pu, W. Zhou, and H. Li, "Dilated convolutional network with iterative optimization for continuous sign language recognition.," in *IJCAI*, pp. 885–891, 2018.

[31] J. Pu, W. Zhou, and H. Li, "Iterative alignment network for continuous sign language recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4165–4174, 2019.

[32] R. Cui, H. Liu, and C. Zhang, "Recurrent convolutional neural networks for continuous sign language recognition by staged optimization," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7361–7369, 2017.

[33] H. Zhou, W. Zhou, and H. Li, "Dynamic pseudo label decoding for continuous sign language recognition," in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1282–1287, IEEE, 2019.

[34] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, "On the properties of neural machine translation: Encoder-decoder approaches," *arXiv preprint arXiv:1409.1259*, 2014.

[35] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[36] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

[37] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision. 2015," *arXiv preprint arXiv:1512.00567*, 2015.

[38] N. B. Ibrahim, H. H. Zayed, and M. M. Selim, "Advances, challenges and opportunities in continuous sign language recognition," *Journal of Engineering and Applied Sciences*, vol. 15, no. 5, pp. 1205–1227, 2020.

[39] F. Quiroga, "Sign language recognition datasets." https://facundoq.github.io/unlp/sign_language_datasets/index.html, 2017. Accessed 10 June 2020.

[40] F. Veiga and A. B. Souza, *Physical Exam Manual*. Elsevier Brasil, 2019.

[41] M. H. Swartz, *medical semiology treatise*. Elsevier Brasil, 2015.

[42] M. Rahman, *Beginning Microsoft Kinect for Windows SDK 2.0: Motion and Depth Sensing for Natural User Interfaces*. Apress, 2017.

[43] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[44] J. Brownlee, *Deep Learning for Computer Vision: Image Classification, Object Detection, and Face Recognition in Python*. Machine Learning Mastery, 2019.

[45] T. E. Oliphant, *A guide to NumPy*, vol. 1. Trelgol Publishing USA, 2006.

[46] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. The MIT Press, 2016.

[47] D. Guo, W. Zhou, H. Li, and M. Wang, "Hierarchical lstm for sign language translation," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[48] J. Huang, W. Zhou, Q. Zhang, H. Li, and W. Li, "Video-based sign language recognition without temporal segmentation," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[49] F. Chollet *et al.*, "Keras." https://keras.io, last accessed on 02/08/21, 2015.

[50] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, *et al.*, "Tensorflow: Large-scale machine learning on heterogeneous distributed systems," *arXiv preprint arXiv:1603.04467*, 2016.

[51] J. Brownlee, *Better Deep Learning: Train Faster, Reduce Overfitting, and Make Better Predictions*. Machine Learning Mastery, 2018.

[52] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.

[53] A. M. Saxe, J. L. McClelland, and S. Ganguli, "Exact solutions to the nonlinear dynamics of learning in deep linear neural networks," *arXiv preprint arXiv:1312.6120*, 2013.

[54] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 249–256, 2010.

[55] G. Hinton, N. Srivastava, and K. Swersky, "Neural networks for machine learning lecture 6a overview of mini-batch gradient descent," *Cited on*, vol. 14, p. 8, 2012.

[56] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[57] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.

[58] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[59] M. D. Zeiler, "Adadelta: an adaptive learning rate method," *arXiv preprint arXiv:1212.5701*, 2012.

[60] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization.," *Journal of machine learning research*, vol. 12, no. 7, 2011.

[61] A. Krogh and J. A. Hertz, "A simple weight decay can improve generalization," in *Advances in Neural Information Processing Systems*, pp. 950–957, 1992.

[62] J. Carreira and A. Zisserman, "Quo vadis, action recognition? a new model and the kinetics dataset," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6299–6308, 2017.

[63] H. Liu, S. Jin, and C. Zhang, "Connectionist temporal classification with maximum entropy regularization," *Advances in Neural Information Processing Systems*, vol. 31, pp. 831–841, 2018.

[64] C. Wu, M. J. Gales, A. Ragni, P. Karanasou, and K. C. Sim, "Improving interpretability and regularization in deep learning," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 2, pp. 256–265, 2017.