Efficient Kinect Sensor-based Kurdish Sign Language Recognition Using Echo System Network

Sami F. Mirza[1], Abdulbasit K. Al-Talabani[2]

[1]Department of Computer Science, Faculty of Science, Soran University, Soran, Kurdistan Region – F.R. Iraq
[2]Department of Software Engineering, Faculty of Engineering, Koya KOY45, Kurdistan Region – F.R. Iraq

***Abstract*—Sign language assists in building communication and bridging gaps in understanding. Automatic sign language recognition (ASLR) is a field that has recently been studied for various sign languages. However, Kurdish sign language (KuSL) is relatively new and therefore researches and designed datasets on it are limited. This paper has proposed a model to translate KuSL into text and has designed a dataset using Kinect V2 sensor. The computation complexity of feature extraction and classification steps, which are serious problems for ASLR, has been investigated in this paper. The paper proposed a feature engineering approach on the skeleton position alone to provide a better representation of the features and avoid the use of all of the image information. In addition, the paper proposed model makes use of recurrent neural networks (RNNs)-based models. Training RNNs is inherently difficult, and consequently, motivates to investigate alternatives. Besides the trainable long short-term memory (LSTM), this study has proposed the untrained low complexity echo system network (ESN) classifier. The accuracy of both LSTM and ESN indicates they can outperform those in state-of-the-art studies. In addition, ESN which has not been proposed thus far for ASLT exhibits comparable accuracy to the LSTM with a significantly lower training time.**

***Index Terms*—Deep learning, Echo system network, Long short-term memory, Microsoft Kinect v2 Sensor, Recurrent neural network, Sign language.**

I. Introduction

In 2020, the World Health Organization (WHO) reported that approximately 466 million people in the world have hearing loss, of whom 34 million are children (World Health Organization, 2020). Those people who lack the ability to listen and\or speak with ordinary people may not be able to understand them. Sign language has therefore become a vital element of human communication. There are now various available sign languages such as American, British, Chinese, Russian, Indian, Persian, and Arabic Sign Language.

Kurdish sign language (KuSL) has been developed only recently. There are three different dialects in Kurdistan for the sign languages studied in deaf private schools in Kurdistan region. KuSL originated in a school for deaf students in Sulaymaniyah in 1982. Students from all Kurdish schools catering for the deaf are able to clearly understand each other using KuSL. In 2015, more than 1000 students attended classes of the deaf schools. The total number of deaf people in Iraqi Kurdistan is estimated to be in excess of 10,000 (Wikipedia, 2019,).

Automatic sign language recognition (ASLR) recognition exploits the physical (dynamic) movement of the hands, face, fingers, or entire body to translate the signs into text or speech. ASLR is a field that has been studied for various sign languages (Almasre and Al-Nuaim, 2016, Maass et al., 2002, Mahmood et al., 2018). However, few research studies have been conducted on the KuSL because it is new (Abdul et al., 2020, Hashim and Alizadeh, 2018, Mahmood et al., 2018).

In this study, we designed a model to translate KuSL into text. There are three main problems that this paper tries to investigate, one of which related to the KuSL itself, where no comprehensive dataset is available with an adequate number of samples. The other two problems related to ASLR are the complexity of the feature extraction step and the high cost of the training stage in the classification level, which negatively influence the ASLT systems' adaptability. Consequently, the questions under investigation in this paper are: (1) How to decrease the time complexity in the feature extraction step, without accuracy degrading of ASLR model? (2) How to design a high-performance model that avoid the time complexity resulted in training the large number of parameters?

In this study, Kinect sensor V2 was used to record videos of seven expert persons expressing 84 different signs (alphabets, digits and words) each with five trials. The features that fed the model were extracted from the videos as a time series representation of only the skeleton points captured by Kinect sensor V2. We have avoided extracting features from the color version of the images to reduce the complexity and proposed a feature engineering applied to the Skelton points instead, to generate more representations of these features. The study has also proposed two recurrent neural network (RNN)-based models: Long short-term memory (LSTM) and

ARO-The Scientific Journal of Koya University
Vol. IX, No.1 (2021), Article ID: ARO.10827, 9 pages
DOI: 10.14500/aro.10827
Received: 05 June 2021; Accepted: 16 August 2021
Regular research paper: Published: 13 October 2021
Corresponding author's e-mail: abdulbasit.faeq@koyauniversity.org

echo system network (ESN). The LSTM, (where weights are learned) are proposed in the literature for ASLR (Liu et al., 2016) and has recorded high performance, however, with a long computation time and complexity. Consequently, we proposed the use of an ESN which has not been used for ASLR so far and uses non-trained weights; therefore, it has a significantly lower time complexity than LSTM; however, we shall see it can achieve comparable accuracy to the LSTM.

The remainder of the paper is structured as follows: Section 2 presents a review of the literature while section 3 explains the background to the study. The adopted methodology is presented in section 4, following which the results are presented and discussed in section 5. Finally, the conclusion to the paper is presented in section 6.

II. Literature Review

In recent years, various studies have been conducted on different sign languages, such as Persian sign language (Karami et al., 2011), American sign language (Truong et al., 2016), Brazilian sign language (BSL) (Dos Santos Anjo et al., 2012), and British sign language (Liwicki and Everingham, 2009, Capilla, 2012).

Sign language recognition is much more challenging where the number of classes (signs) is high. Studies in the literature mainly used a limited number of classes (Table I). For KuSL, and to our best knowledge, the only two available studies have used 12 (Hashim and Alizadeh, 2018) and 10 signs (Mahmood et al., 2018). In this study, and to increase the validity of the proposed ASLR for KuSL, we have designed a dataset that contains 84 signs including digits, alphabets and words. In addition, we have conducted our proposed model on the Chinese Sign Language dataset (Liu et al., 2016), with 100 classes for further validation step.

The nature of the application whether it deals with static (represented in one image) or with dynamic (represented in a sequence of images) language signs, in one side and/or with discrete versus continuous input in the other side, controls the method of data collection, feature extraction, and the classification. Static signs can be easily represented in a global feature, because it has not a time series nature, for example: (Karami et al., 2011) investigated the recognition of static gestures for the Persian alphabet expressed in sign language. Overall, 32 alphabet static signs were recorded using a digital camera and a multilayer perceptron NN (MLP_NN) classifier was utilized to train the proposed model. The study achieved an accuracy of 94%. In another study of BSL, (Dos Santos Anjo et al., 2012) explored the recognition of static gestures in BSL using depth information extracted by Kinect Xbox 360. The authors worked on ten alphabets and applied image segmentation and classification using an artificial NN (ANN). The accuracy of these models was 75.4% for MLP and 67.47% for segmentation.

Regarding the dynamic signs, features should be extracted from each time step, since global features for time series-based signals may lead to lose information of sequences. Therefore, Capilla, 2012, proposed the use of dynamic time warping and utilized Kinect Xbox to translate 14 sign words and achieved an accuracy of 95.2%. In the other side, Truong et al., 2016, proposed the translation of American sign language for alphabetical text and speech using both static and dynamic inputs. ASL alphabets consist of 26 letters, 24 of which are represented statically and 2 signs (“J” and “Z”) are represented dynamically, therefore requiring gestures to be expressed. The authors used Logitech webcam to collect data and adopted two types of classification: Adaboost and Haar_Like classifiers, and they achieved accuracy of 98%.

There are three types of sign language features: Hand motion, hand position, and hand shape, and these have been

Table I
Summary of the datasets used in the literature.

Ref.	No. of Sign	No. of subjects	No. of Samples	Method	accuracy %
(Hashim and Alizadeh, 2018)	12	-		Enhancement and Segmentation	67
(Mahmood et al., 2018)	10	10	200	MLP	98
(Karami et al., 2011)	32	-	640	MLP	94
(Truong et al., 2016)	26		28000 and 11100	Adaboost, Haar_Like classifier	98
(Dos Santos Anjo et al., 2012)	10		400	MLP, Virtual Wall and Libras specific	75
(Liwicki and Everingham, 2009)	100		1000	Robust, Bootstrap and HMM	
(Capilla, 2012)	14		70	Nearest group and DTW	95.2
(Lang et al., 2012)	25			HMM and Dragonfly NITE	97
(Preeti Amatya and Gerrit Meixner, 2018)	11			DTW and VGB	65
(Chai et al., 2013)	239		1195	Kinect v2	83.5
(Kumar et al., 2018)	30	10	2700	HMM and SVM	83.7
(Almasre and Al-Nuaim, 2016)	28	4	224	Supervised learning-	-
(Li, 2012)	9	4	3600	K-Mean and Graham Scan	91
(El-Bendary et al., 2010)	15		15	MLP and MD	91.3
(Mittal et al., 2019)	35	6	3150	CNN	89.5
(Liu et al., 2016)	100 500	50 50	25,000 125,000	RNN and LSTM	86
(Lee et al., 2021)	26	100	2600	LSTM and k-Nearest-Neighbor	91.8
(Rastgoo et al., 2020)	100 45 36	10 25	10,000 100 K 81,000	Single shot detector, 2D Convolutional NN, 3D Convolutional NN, and LSTM	91
(Li et al., 2020)	2000	100	34,404	Holistic visual appearance based approach, and 2D human pose-based approach	62.6
(Gao et al., 2021)	100	50	25,000	H2SNet	0.91
(Katılmış and Karakuzu, 2021)	50	4	8000	ML-KELM	98

MLP: Multilayer perceptron, LSTM: Long short-term memory

adapted to translate Japanese sign language (Awata et al., 2017, Lee et al., 2016). We have adopted in this study the use of the skeleton position alone to reduce the complexity of feature extraction step.

Feature for time series sign language may suffer from the poor representation and\or high dimensionality and lead to increase of the classification complexity. For example, a dataset was designed using 80 Chinese sign language words and Kinect v2 was employed to exploit the RGB image, depth map, and position of the skeleton joints (Li et al., 2017). In ddition, Li et al., 2020, proposed a new large-scale word-level American sign language video dataset, and used two holistic visual appearance based approach, and 2D human pose-based approach models. The utilized dataset was collected from YouTube and contain more than 2000 words performed by 100 signers. The study achieves an accuracy of 62.63%.

In Li et al., 2021, authors present a multiscale fully convolutional NN (MFCN) based method, and extracted the detailed features of the ground object using multiscale convolution kernels. Resolving the findings of change detection (CD) can also be harmed by an imbalance of positive and negative samples. And then model has been trained by unbalanced samples. Hence, using digital globe dataset, the suggested technique was compared to six state-of-the-art CD methods. Finally, their research shows that the achieved accuracy for proposed method is higher than state of the art methods. In (Hossein and Ejaz, 2020), authors proposed a deep convolutional NNs to learn on images of Bengali sign language. Dataset has been collected by snapshot from video using webcam and then applied computer vision-based method. The dataset includes ten set of images and used 31 different signs and the total number of images are 310 Bangladesh image signs. The research achieved an accuracy of 99.8%.

To avoid using the full image-based feature recent studies utilized some devices as a feature extractor. In 2011, Microsoft produced a new device to record video and capture images called the Kinect v2 sensor, which provides RGB, Depth sensor, and 3D Skeleton (Chai et al., 2013). (Almasre and Al-Nuaim, 2016) used depth sensors in Kinect with HMM to recognize gestures and achieved an accuracy of 83.7%. In another study (Kumar et al., 2018), adopt depth sensor data to recognize hand gestures. Lang et al., 2012, and Preeti Amatya and Gerrit Meixner, 2018, propose a dynamic translation of hand gestures to text, where the data has been collected using Kinect v2 and SDK2.0 software. And most recently, leap motion device has been used on a system that is tested with 942 signed sentences using 35 different sign words of Indian Sign Language. The average accuracy of 72.3% and 89.5% has been achieved on signed sentences and isolated sign words, respectively, (Mittal et al., 2019). Kratimenos et al., 2021, employed a SMPL-X model (an extension from the Skinned Multi-Person Linear Model) to enable and extract features from hand, body and face in one RGB image and used to SLR.

Effective and up-to-date classifier for time series inputs are the RNN-based models. For example, an LSTM- based model is proposed in several projects (for example, Li et al., 2017; Liu et al., 2016) using leap motion sensors and color images in Kinect v2. In addition, Li et al. (2017) propose specific hand shape as a descriptor for hand shape, and achieve better sign recognition results, when applied to a proposed encoder-decoder LSTM model. Lee et al., 2021, provides a prototype for an ASL learning application. They applied two methods which are LSTM and k-Nearest-Neighbor for 26 ASL alphabets expressed by 100 subjects each with 100 trails, and they achieved an accuracy of 91.8%. A novel deep learning-based pipeline architecture is proposed by (Rastgoo et al., 2020) for efficient automatic hand sign language recognition from RGB input videos, based on the single shot detector, 2D convolutional NN, 3D convolutional NN, and LSTM, and dataset has been containing 10,000 RGB video from 100 Persian sign words. The achieved accuracy is 90% and 91.1% for RKS-Persian dataset and NYU dataset respectively.

However, training of LSTM as an RNN-based model is reported to be inherently difficult (Lukoševičius, 2012). As a consequent, in this paper, we have proposed ESN architecture as a new powerful approach in RNN research, where, instead of difficult learning process, it based on the property of untrained randomly initialized RNN (Čerňanský and Tiňo, 2007). We shall see in section 5 that the ESN can significantly reduce the training time and achieve results comparable to the LSTM.

Finally, and regarding the ASLR applied to KuSL, two studies have been conducted using image processing tools. First, Hashim and Alizadeh, 2018, developed an algorithm using a grid-based gesture descriptor on the hand gesture image for 12 Kurdish letters, produced following image enhancement and segmentation steps. The achieved accuracy of the proposed model was 67%. In the second study by Mahmood et al., 2018, ten words were expressed by ten people using KuSL. The classifier used was ANN_MLP and the dataset consisted of 200 images extracted from frames 16 and 30 only. The model accuracy was reported to be 98%.

Table I summarizes the datasets of some of the papers produced in sign language.

III. Background

A. Feature Extraction

In computer vision, feature extraction is utilized to capture “important” information on the selected body joints or the detection of hand gestures using image processing tools (Gilorkar and Ingle, 2014). Features can be extracted in different ways such as hand segmentation using canny edge detection (Prasad et al., 2016), scale-invariant feature transform (Pandita and Narote, 2013), Haar wavelets, Haar-like features (Chen et al., 2008), Fourier descriptors, and using Microsoft Kinect Sensor V1 and V2 (Almasre and Al-Nuaim, 2016, Awata et al., 2017, Capilla, 2012, Chai et al., 2013, Kumar et al., 2018, Lang et al., 2012, Preeti Amatya and Gerrit Meixner, 2018, Verma et al., 2013).

In this work, we used Microsoft Kinect sensor V2 to extract 25 joints from the body (Fig. 1). The sensor computes

http://dx.doi.org/10.14500/aro.10827

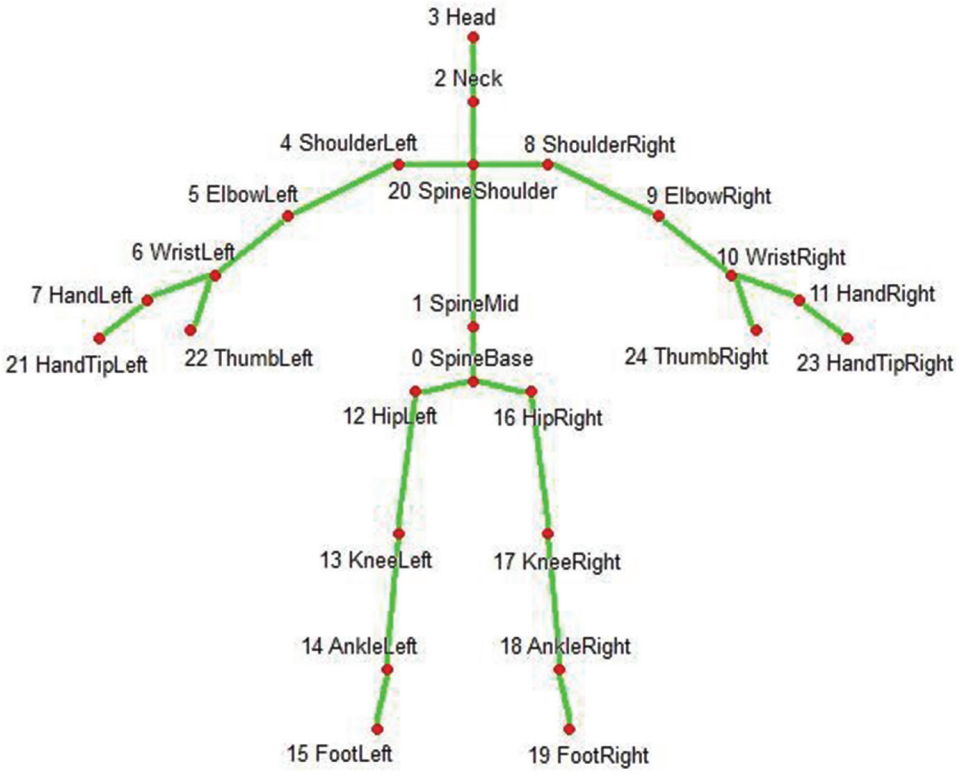

Fig. 1. Skeleton joints captured by Kinect sensor V2.

the joint position values (x, y, and z). In the current study, we extracted 15 joints representing the upper part of the body where most relevant information to the sign language is available. The extracted joints were six joints for each hand (Shoulder, Elbow, Wrist, Hand, Hand Tip, and Thumb) along with the Neck, Head, and Spine Shoulder joints. In addition, a feature engineering approach was adopted that involved determining the slope of each feature along the time steps denoted in this work as a delta feature (DF).

B. LSTM

The problem of gradient vanishing in RNN means alternative models is required. The LSTM is one of the proposed models that aim to improve the RNN. The network structure of the LSTM is complicated. The main improvement LSTM offers over RNN is its ability to capture long-term dependencies (Prabakaran and Shyamala, 2019). Therefore, LSTM adopts a structure that can overcome the problem of gradient vanishing. The LSTM is one of the deep learning classifiers that deal with time-series data such as video, voice, and vibration (Abdul et al., 2020). It takes account of the diversity of the multidimensional feature values at each time step. There are four parts in each repeating module: cell state (c), input gate (i), forget gate (f), and output gate (o) (Fig. 2).

The cell value computation at the current time c_t is represented in equation 1:

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \quad (1)$$

Where f_t is the forget gate at time t, c_{t-1} is the state of the previous cell, $\odot$ refers to an element wise multiplication, i_t is the input gate at time t, and g_t is computed using equation 4. The function of the cell state is to remember a value during the recurrent connection.

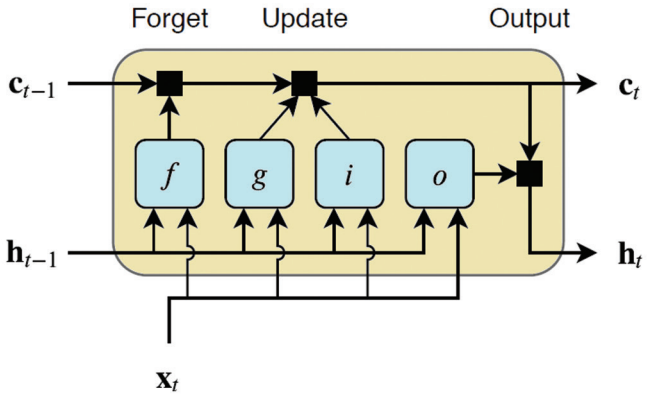

Fig. 2. The structure of the long short-term memory cell.

The following equation demonstrates how to compute, f_t, i_t, and g_t:

$$f_t = \sigma_g \left(W_f x_t + R_f h_{t-1} + b_f \right) \quad (2)$$

$$i_t = \sigma_g \left(W_i x_t + R_i h_{t-1} + b_i \right) \quad (3)$$

$$g_t = \sigma_c \left(W_g x_t + R_g h_{t-1} + b_g \right) \quad (4)$$

Where σ_g and σ_g are gate activation functions, W is the input weight, R represents the recurrent weight, and b is the bias of each component. It is also important to note that the hidden state is updated using equation (5):

$$h_t = o_t \odot \sigma_g\left(c_t\right) \tag{5}$$

Where:

$$o_t = \sigma_g\left(W_o x_t + R_o h_{t-1} + b_o\right) \tag{6}$$

crucial step in the learning process of the model consists of remembering and then forgetting the values. The updated inputs remember values in the memory whereas the forgetting gates skip the remembered input when it is no longer important. The output gate determines when the cell state produces the output value. The output of the final steps during the computation at each gate and cell state forms the input of the later steps. This enables the LSTM model to learn how to maintain its memory as a function of previous values (Jena et al., 2014).

C. Echo State Network (ESN)

ESN provide a supervised learning architecture for RNNs. It adopts a random and non-trained RNN with the input signal, thereby inducing in each neuron within this "reservoir" network a nonlinear response signal, additionally, it combines a desired output signal by a trainable linear combination of all of these response signals.

RNN is traditionally represented by the equation:

$$h(t) = f\left(x(t), h(t-1); \theta_{enc}\right) \tag{7}$$

Where $h(t)$ and $h(t\text{-}1)$ are the current and previous states, respectively, $x(t)$ is the input, $f(t)$. Is the non-linear activation function, and θ_{enc} denotes the trainable encoding parameters. Equation (7) can be rewritten as:

$$h(t) = tanh\left(W_{in} x(t) + W_r h(t-1)\right) \tag{8}$$

Thus:

$$\theta_{enc} = \left\{W_{in}, W_r\right\} \tag{9}$$

Where the matrices W_{in} and W_r are the weights of the input and recurrent connections, respectively. The collection of all states, h, is represented H= [h(1), h(2),…(h(T)], where T is the number of time steps in the sample x. To make this representation suitable for different classifiers, one representation of H can be adopted, denoted here as r(H). A possible representation r is equal to $h(T)$.

To perform the classification step, a function g(.) takes the output of the representation as an input and then maps the representation to one of the categories.

$$y = g\left(r_X; \theta_{dec}\right) \tag{10}$$

Where θ_{dec} denotes all trainable parameters in the classifier.

The traditional RNN trains all the parameters θ_{enc} and θ_{dec}. To avoid the high computational complexity of back-propagating through time, the reservoir computing approach generates the θ_{enc} weights randomly.

However, this may result in a lack of adaptability. To resolve this problem, a large recurrent layer can be used to make the reservoir generate a rich pool of diverse dynamics to model various tasks.

The reservoir capability of generalization can be improved through the processing units in the recurrent layer, the sparsity of the recurrent connections, and the spectral radius of the connection weights matrix W_r, which is set to bring the system close to stability (Bianchi et al., 2016).

The main hyper parameters that control the behavior of the reservoir are the spectral radius; the percentage of non-zero connections; the number of hidden units R; and the scaling ω of the values in W_{in} (Livi et al., 2017).

IV. Method and Materials

A. Datasets

One of the contributions of this work is to design a specific dataset for KuSL. Here, we focused on skeleton data extracted using Kinect sensor and collected metadata from the camera. The Kinect sensor v2 has two sensors for RGB-color video recording with a resolution of 1920 × 1080 pixels, the depth sensor records video at a resolution of 512 × 424 pixels (Wasenmüller and Stricker, 2016) and 3D types detect 25 joints of the body, with each joint represented in three dimensions, x, y, and z (also known as Skeleton values). This work uses the skeleton-based features. We selected 15 features, when for each one a two-dimensional position is provided. The whole dataset consists of 2940 samples, including 84 classes (35 alphabetic, 39 words, and 10 numbers [0–9]) (Table II). In this dataset, seven professional

Table II
Class label for each of Kurdish (alphabetic, words, and numbers)

Label	class	Label	class	Label	class
1	ئ	13	ها	25	ق
2	ه	14	ح	26	ر
3	ع	15	ج	27	ڕ
4	ا	16	ک	28	س
5	ب	17	خ	29	ش
6	چ	18	ل	30	ت
7	د	19	لا	31	ف
8	ی	20	ڵ	32	و
9	ێ	21	م	33	وو
10	ف	22	ن	34	ز
11	گ	23	ۆ	35	ژ
12	غ	24	پ		
36	بەڵێ	50	گران بەها	64	سارد
37	ببورە	51	هەرزان	65	سەرچاو
38	بەیارمەتیت	52	هەینی	66	سێ شەم
39	بەرامبەر	53	جوان	67	شەممە
40	بەیانی	54	کات	68	سڵاو
41	چاکەت	55	کورت	69	سپی
42	چاڵاک	56	لە تەنیشت	70	سوور
43	چۆنی	57	من	71	خواحافیز
44	چوارشەم	58	ناوت چیە	72	یەک شەم
45	دەزانم	59	نەخێر	73	زانا
46	درێژ	60	نازانم	74	زیرەک
47	دووشەم	61	نێوان		
48	ئێمە	62	پێنج شەم		
49	گەرم	63	رۆژئاوابوون		
75	سفر	79	چوار	82	حەوت
76	یەک	80	پێنج	83	هەشت
77	دوو	81	شەش	84	نۆ
78	سێ				

http://dx.doi.org/10.14500/aro.10827

subjects were involved to express the signs and to record the videos, where their skeleton data has been captured. Each subject has repeated the same sign 5 times. Hence, for each class (7×5=35) samples are collected, whereas the total number of samples is 7×5×84=2940 samples.

Kurdish letters can be expressed in both dynamic and static ways. However, most of the letters are static (with no movement), just two of them are dynamic, such as (ۆ، ێ), Fig. 3. However, there are many dynamic signs in the adopted dataset in this work.

B. Feature Engineering (Delta)

Feature engineering can be applied when new features need to be produced from those available. One of the features adopted in this work was the delta of the joint positions along the time steps. The delta representation of a feature is the difference between the value of a feature in the next time step and the value of the same feature in the current time step. In this project, 15 joints were used, which were represented in two dimensions (x, y). Completing all of these computations resulted in 30 DFs, with a reduced o1-time step. To reduce the resolution and hence the computation of the samples, the time steps with odd indexes were included in the delta computation.

C. Scale Normalization

Recording videos for each participant using Kinect V2 led to some differentiating parameters. One of which was the distance from the camera to the position of the participant. The adopted camera detects a person from 1 m to 4.5 m away, and may capture images at different distances from the participant. For automatic scaling, a normalization (standardization) approach using mean and standard deviation was applied, as shown in equation (11):

$$z = \frac{x - \mu}{\sigma} \tag{11}$$

Where μ and σ are the mean and the standard deviation of the feature values in one frame.

D. The LSTM Designed Model

This work made use of two models. The first is the BiLSTM model, the structure of which consists of two BiLSTM layers (sequence to sequence and sequence to label) with the number of neurons equal to 125 and 100, respectively. In addition, we used one fully connected layer with 84 nodes (Fig. 4).

E. The ESN Model

For this work, we adopted the model developed by Bianchi et al., 2020, which is depicted in Fig. 5. Here, we employed a bi-directional representation of the signal that fed a reservoir, producing a collection of states denoted as H. High dimensional time series data significantly increases the complexity. Therefore, the dimension of $\check{H}$ was reduced using principal component analysis (PCA) to produce a low dimensional representation denoted as $\check{H}$. However,

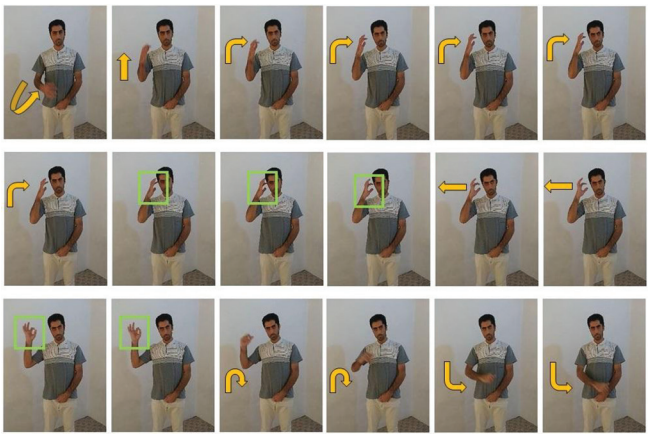

Fig. 3. A Kurdish dynamic sign representing the word “Zirak” meaning brave.

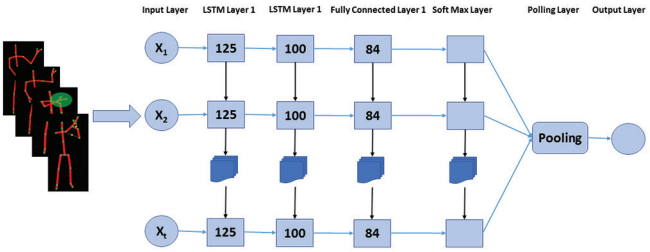

Fig. 4. Long short-term memory model structure.

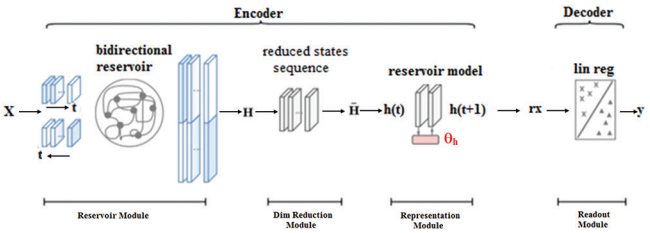

Fig. 5. Echo state network model structure (Bianchi et al., 2020).

$\check{H}$ still has two dimensions (features and time steps), thus needs to be represented in one dimension to be useful for the majority of the machine learning techniques. We have adopted a reservoir model (Bianchi et al., 2020) to produce a one-dimensional representation of $\check{H}$ to later feed the linear readout classifier. In a standard ESN, the readout is linear and is quickly trained by solving a convex optimization problem. The most important advantage of ESN that is adopted here is the low complexity because of the untrained architecture. Therefore, unlike the LSTM, we shall see in the next section how the ESN will significantly reduce the training time, with a very close accuracy to the LSTM.

V. Results and Discussion

A. LSTM Model

The experiments utilized different features for the BiLSTM, including the original feature (OF) (30 skeleton position), the Delta of the Features (DF) (30 feature), and their combination (CF) (60 features). It is important to note that the computation time using CF is extremely high;

consequently, we adopted the low resolution (LR) form of the frame representation by ignoring frames with even indexes. The normalization step was also investigated in all the experiments. To validate the parameters of the Bi-LSTM, we adopted a cross-validation approach (10% vs. 70% for validation and training, respectively, and 20% for testing the model) to tune parameters such as the number of epochs and mini-batch size.

The validation results indicate that large number epochs and small mini-batch sizes achieve better results when using our designed Kurdish dataset. Based on the available hardware in this study and to avoid extremely lengthy computation time, the optimum number of epochs in this model was found to be 250 and the batch size was 1. The small batch size may reflect the diversity of the data, which could be due to the limited number of participants and the number of samples per class in the adopted dataset. Table III displays the accuracy obtained by the BiLSTM classifier using different features. Although the results for all the experiments are similar, for 588 samples the adopted normalization step and the delta computation demonstrated a significant ability to increase the accuracy of the results. The highest accuracy (98.5%) was achieved by the normalized combination of the original and its DFs. The normalization step improved the accuracy of both OF and CF-based models, but was not able to improve the DF-based model. This could be due to the scaling that took place during the delta computation.

B. ESN Model

For the ESN model, we carried out the same experiments while using CF features without decreasing the resolution. This is because the time complexity of ESN is significantly less than that of the LSTM. In the ESN model, the main challenge lies in the parameter validation step as ESN is unstable due to the randomness of the weight initialization. However, adopting the same cross-validation approach tuned the size of the reservoir to be 590, the largest eigenvalue of the reservoir (spectral radius) as 0.2, the amount of leakage in the reservoir state update as 0.6, the percentage of nonzero connections in the reservoir (connectivity) as 0.10, the scaling of the input weights to be 0.3, the noise in the reservoir state update as 0.01, the number of transient states to be dropped as 1, and the number of epochs to be 1000. The achieved results for the ESN mode are presented in Table IV.

Unlike the LSTM, the normalization step in the ESN has not improved the accuracy of OF and CF; however, it was the combination feature in both versions that yielded the highest accuracy. The normalization pre-processing in the PCA included in the proposed ESN model therefore appeared to have an effect on the performance of the model.

It is worthy to highlight the significant lower computation time of the ESN compared to the LSTM (Fig. 6); however, a comparable accuracy has been achieved by the ESN (Fig. 7). The states in ESN are transformed into the random weight space and seem to produce good representation of the model. This may have a link to various studies that

Table III
Different feature representation with BiLSTM, where the number of epochs is 250 and batch size = 1

No	Feature	Accuracy (%)	Time (Minutes)
1	Original feature	97.7	925
2	Delta feature	98.1	912
3	Combination feature (low resolution)	96.1	488
4	The normalized original feature	98.3	939
5	The normalized delta feature	97.8	907
6	The normalized combination feature (low resolution)	98.5	

Table IV
Accuracy of ESN using different feature representation

No	Feature	Accuracy (%)	Time (Minutes)
1	Original feature	97.3	6
2	Delta feature	95.6	10
3	Combination feature	97.1	7
4	Combination feature (low resolution)	97.4	1
5	The normalized original feature	94.4	4
6	The normalized delta feature	96.4	6
7	The normalized combination feature	93.5	4
8	The normalized combination feature (low resolution)	91	1

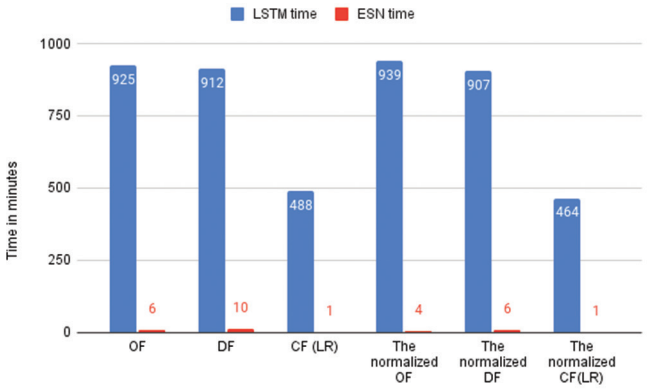

Fig. 6. Long short-term memory and echo state network training time for different set of features.

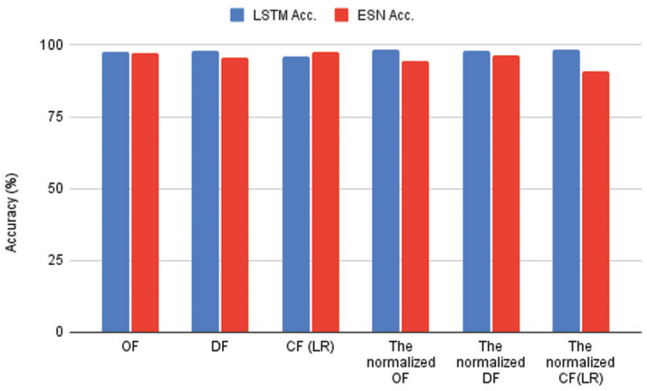

Fig. 7. Long short-term memory and echo state network accuracies for different set of features.

prove the usefulness of the random projection in many high dimensional space produced by the states of the ESN (Al-Talabani et al., 2015).

C. State of the Art Studies

In state-of-art studies, several strands of research have been conducted on a variety of styles of sign language. However, to make a comparison with our model, the dataset needs to have the skeleton version of the data available. One of the most well-known datasets which provide the skeleton version of the data and is available online is the Chinese Sign Language dataset (Liu et al., 2016). This study employed 100 Classes from this dataset, which includes 25000 samples, and adopted the use of the LSTM classifier, achieving 85% accuracy. By applying the adopted BiLSTM model to the same dataset, the accuracy increased to 95%. Similarly, when the ESN model was used, 94.6% accuracy was achieved. Additionally, in comparison with the results reviewed in Table I, we conclude that none of the reviewed works exceed our proposed BiLSTM model as the best result was 98% whereas the proposed BiLSTM achieves 98.5%. It is important to emphasize that the number of signs involved in each of the datasets presented in Table I are all smaller than our proposed dataset with the exception of (Liwicki and Everingham, 2009), which included 230 signs. However, the latter study achieved an accuracy of 83%, which is significantly less than our proposed model. It is also worthy to highlight that the proposed ESN model achieves an accuracy of (97.4%), which is comparable to the achieved results in the state-of-the-art studies, however, with significantly lower complexity.

VI. Conclusion

The skeleton points extracted by the Kinect sensor V2 have a strong ability to capture sign language-related information. Furthermore, aspects of feature engineering, such as computing the delta of the positions of each skeleton, can add complementary information to the features of skeleton positions. The nature of gesture data is that it is a time-varying signal represented in the sequence of the video frames. As it is well-known, RNN based models, especially the BiLSTM, achieve outperforming accuracy for ASLR. However, the training stage for the BiLSTM is time-consuming and may take many hours. As an alternative to reduce the complexity of the model, the ESN (where no trained weights are adopted) can achieve comparable performance and a highly significant decrease in computation time.

The high performance of ESN highlights the usefulness of transforming the gesture data on the random weight vectors adopted in the ESN. For the future, this motivates us for further investigation to link this to the random projection capability to extract valuable information for gestures. User independent ASLR, where the samples tested for a subject is not available in the training stage, can also be studied to more validation the current work.

References

Abdul, Z.K., Al-Talabani, A.K. and Ramadan, D.O., 2020. A hybrid temporal feature for gear fault diagnosis using the long short term memory. *IEEE Sensors Journal*, 20(23), pp.14444-14452.

Almasre, M.A. and Al-Nuaim, H., 2016. A real-time letter recognition model for arabic sign language using Kinect and leap motion controller v2. *International Journal of Advanced Engineering, Management and Science*, 2(5), p.239469.

Al-Talabani, A., Sellahewa, H. and Jassim, S.A., 2015. Emotion recognition from speech: Tools and challenges. Mobile multimedia/image processing, security, and applications. *International Society for Optics and Photonics*, 9497, p.94970N.

Awata, S., Sako, S. and Kitamura, T., 2017. Japanese sign language recognition based on three elements of sign using kinect v2 sensor. In: *International Conference on Human-computer Interaction*. Springer, Berlin, Germany, pp.95-102.

Bianchi, F.M., Livi, L. and Alippi, C., 2016. Investigating echo-state networks dynamics by means of recurrence analysis. *IEEE Transactions on Neural Networks and Learning Systems*, 29(2), pp.427-439.

Bianchi, F.M., Scardapane, S., Løkse, S. and Jenssen, R., 2020. Reservoir computing approaches for representation and classification of multivariate time series. *IEEE Transactions on Neural Networks and Learning Systems*, 23(5), pp.2169-2179.

Capilla, D.M., 2012. *Sign Language Translator Using Microsoft Kinect Xbox 360 tm*. Department of Electrical Engineering and Computer Science, University of Tennessee, Tennessee.

Čerňanský, M. and Tiňo, P., 2007. Comparison of echo state networks with simple recurrent networks and variable-length Markov models on symbolic sequences. In: *International Conference on Artificial Neural Networks*. Springer, Berlin, Germany, pp.618-627.

Chai, X., Li, G., Lin, Y., Xu, Z., Tang, Y., Chen, X. and Zhou, M., 2013. Sign language recognition and translation with kinect. *IEEE Conference on Automatic*, 655, p.4.

Chen, Q., Georganas, N.D. and Petriu, E.M., 2008. Hand gesture recognition using Haar-like features and a stochastic context-free grammar. *IEEE Transactions on Instrumentation and Measurement*, 57(8), pp.1562-1571.

Dos Santos Anjo, M., Pizzolato, E.B. and Feuerstack, S., 2012. *A Real-time System to Recognize Static Gestures of Brazilian Sign Language (Libras) Alphabet Using Kinect*. IHC, Citeseer, pp.259-268.

El-Bendary, N., Zawbaa, H.M., Daoud, M.S., Hassanien, A.E. and Nakamatsu, K., 2010. Arslat: Arabic sign language alphabets translator. In: *International Conference on Computer Information Systems and Industrial Management Applications*. IEEE, United States, pp.590-595.

Gao, L., Li, H., Liu, Z., Liu, Z., Wan, L. and Feng, W., 2021. RNN-transducer based Chinese sign language recognition. *Neurocomputing*, 434, pp.45-54.

Gilorkar, N.K. and Ingle, M.M., 2014. A review on feature extraction for Indian and American sign language. *International Journal of Computer Science and Information Technologies*, 5(1), pp.314-318.

Hashim, A.D. and Alizadeh, F., 2018. Kurdish sign language recognition system. *UKH Journal of Science and Engineering*, 2(1), pp.1-6.

Hossein, M.J. and Ejaz, M.S., 2020. Recognition of Bengali sign language using novel deep convolutional neural network. In: *2nd International Conference on Sustainable Technologies for Industry 4.0 (STI), 2020*. IEEE, United States, pp.1-5.

Karami, A., Zanj, B. and Sarkaleh, A.K., 2011. Persian sign language (PSL) recognition using wavelet transform and neural networks. *Expert Systems with Applications*, 38(3), pp.2661-2667.

Katilmis, Z. and Karakuzu, C., 2021. ELM based two-handed dynamic turkish sign language (TSL) word recognition. *Expert Systems with Applications*, 2021, pp.115213.

Kratimenos, A., Pavlakos, G. and Maragos, P., 2021. Independent sign language recognition with 3d body, hands, and face reconstruction. In: *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, United States, pp.4270-4274.

Kumar, P., Saini, R., Roy, P.P. and Dogra, D.P., 2018. A position and rotation invariant framework for sign language recognition (SLR) using Kinect. *Multimedia Tools and Applications*, 77(7), pp.8823-8846.

Lang, S., Block, M. and Rojas, R., 2012. Sign language recognition using kinect. In: *International Conference on Artificial Intelligence and Soft Computing*. Springer, Berlin, Germany, pp.394-402.

Lee, C.K., Ng, K.K., Chen, C.H., Lau, H.C., Chung, S. and Tsoi, T., 2021. American sign language recognition and training method with recurrent neural network. *Expert Systems with Applications*, 167, pp.114403.

Lee, G.C., Yeh, F.H. and Hsiao, Y.H., 2016. Kinect-based Taiwanese sign-language recognition system. *Multimedia Tools and Applications*, 75(1), pp.261-279.

Li, D., Rodriguez, C., Yu, X. and Li, H., 2020. Word-level deep sign language recognition from video: A new large-scale dataset and methods comparison. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*. WACV, United States, pp.1459-1469.

Li, X., He, M., Li, H. and Shen, H., 2021. *A Combined Loss-based Multiscale Fully Convolutional Network for High-resolution Remote Sensing Image Change Detection*. IEEE Geoscience and Remote Sensing Letters, United States.

Li, X., Mao, C., Huang, S. and Ye, Z., 2017. Chinese sign language recognition based on shs descriptor and encoder-decoder lstm model. In: *Chinese Conference on Biometric Recognition*. Springer, United States, pp.719-728.

Li, Y., 2012. Hand gesture recognition using Kinect. In: *2012 IEEE International Conference on Computer Science and Automation Engineering*. IEEE, United States, pp.196-199.

Liu, T., Zhou, W. and Li, H., 2016. Sign language recognition with long short-term memory. In: *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, United States, pp.2871-2875.

Livi, L., Bianchi, F.M. and Alippi, C., 2017. Determination of the edge of criticality in echo state networks through Fisher information maximization. *IEEE Transactions on Neural Networks and Learning Systems*, 29(3), pp.706-717.

Liwicki, S. and Everingham, M., 2009. Automatic recognition of fingerspelled words in british sign language. In: *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*. IEEE, United States, pp.50-57.

Lukoševičius, M., 2012. A practical guide to applying echo state networks. In: *Neural Networks*: *Tricks of the Trade.* Springer, Berlin, Germany.

Maass, W., Natschläger, T. and Markram, H., 2002. Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation*, 14(11), pp.2531-2560.

Mahmood, M.R., Abdulazeez, A.M. and Orman, Z., 2018. Dynamic hand gesture recognition system for kurdish sign language using two lines of features. In: *2018 International Conference on Advanced Science and Engineering (ICOASE)*. IEEE, United States, pp.42-47.

Mittal, A., Kumar, P., Roy, P.P., Balasubramanian, R. and Chaudhuri, B.B., 2019. A modified LSTM model for continuous sign language recognition using leap motion. *IEEE Sensors Journal*, 19(16), pp.7056-7063.

Pandita, S. and Narote, S., 2013. Hand gesture recognition using SIFT. *Proceedings of the IEEE International Conference on Computational Intelligence and Security*, 2(1), p.4.

Prabakaran, D. and Shyamala, R., 2019. A review on performance of voice feature extraction techniques. In: *2019 3rd International Conference on Computing and Communications Technologies (ICCCT)*. IEEE, United States, p.221-231.

Prasad, M., Kishore, P., Kumar, E.K. and Kumar, D.A., 2016. Indian sign language recognition system using new fusion based edge operator. *Journal of Theoretical and Applied Information Technology*, 88(3), p.574-583.

Preeti Amatya, K.S. and Meixner, G., 2018. Translation of sign language into text using kinect for windows v2. In: *The Eleventh International Conference on Advances in Computer-human Interactions.* ACHI, United States.

Rastgoo, R., Kiani, K. and Escalera, S., 2020. Hand sign language recognition using multi-view hand skeleton. *Expert Systems with Applications*, 150, p.113336.

Truong, V.N., Yang, C.K. and Tran, Q.V., 2016. A translator for American sign language to text and speech. In: *2016 IEEE 5th Global Conference on Consumer Electronics*. IEEE, United States, pp.1-2.

Verma, H.V., Aggarwal, E. and Chandra, S., 2013. Gesture recognition using kinect for sign language translation. In: *2013 IEEE 2nd International Conference on Image Information Processing (ICIIP-2013)*. IEEE, United States, pp.96-100.

Wasenmüller, O. and Stricker, D., 2016. Comparison of kinect v1 and v2 depth images in terms of accuracy and precision. In: *Asian Conference on Computer Vision, 2016*. Springer, United States, pp.34-45.

Wikipedia., 2019. *Kurdish Sign Language*. Available from: https://www.en.wikipedia.org/wiki/Kurdish_Sign_Language. [Last accessed on 2019 Mar 28].

World Health Organization., 2020. *Deafness and Hearing Loss*. Available from: https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss. [Last accessed on 2020 Mar 01].