International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Enhancing Sign Language Recognition: A Fusion of Bidirectional LSTMs and BiGRUs in Video Processing

Ajay M. Pol[*1,2], Shrinivas A. Patil [3]

Submitted: 27/01/2024 **Revised**: 05/03/2024 **Accepted**: 13/03/2024

Abstract: In this comprehensive study, the focus is on sign language recognition using the WLASL dataset, where various models undergo evaluation, leading to a meticulous analysis of their performance metrics. Notably, the proposed reinforcement learning (RL) model emerges as a standout performer, showcasing exceptional results with an accuracy rate of 99%, sensitivity at 99%, specificity reaching 98%, and an impressive F1 Score of 99%. A noteworthy observation is the superior feature extraction capabilities of EfficientNet-B1, outperforming the widely used ResNet-101. The integration of bidirectional recurrent neural networks (RNN) emphasizes the critical role of temporal understanding in enhancing the accuracy of sign language recognition. Moreover, the RL-enhanced EfficientNet-B1 demonstrates excellence not only in accuracy but also in generating contextually rich captions, as evidenced by a commendable BLEU score of 0.51. These findings not only contribute significantly to the ongoing advancements in sign language recognition technology but also underscore the pivotal role of reinforcement learning and model selection in achieving heightened accuracy and contextual understanding, particularly within the challenging context of the WLASL dataset.

Keywords: *Bidirectional RNN, Deep Learning, ResNet101, Sign Language Recognition, Video Sequence.*

1. Introduction

The field of sign language recognition from video has become a focal point of research, serving as a pivotal means to bridge communication gaps between the deaf and hearing communities. Recent strides in computer vision and deep learning have paved the way for the development of more accurate and efficient sign language recognition systems [1]. This study delves into the intricate domain of sign language recognition, specifically concentrating on the analysis of video sequences to interpret and translate sign gestures into meaningful text.

Video sequence based sign language recognition is challenging task. The hand gesture based sign language recognition from dynamic video sequences, being inherently dynamic, presents distinct challenges compared to static image recognition, as it captures the temporal evolution of signs. The overarching goal is to create a robust system capable of accurately recognizing and translating a diverse range of sign gestures into text. For establishing the bridge the communication gap between normal and deaf people this work shows its significance beyond technical aspects. The advancement of sign language recognition from video, this research seeks to foster inclusivity, empower individuals with hearing impairments, and promote equal participation in various aspects of life [2].

The proposed system employs a deep learning approach to address the complexities associated with sign language recognition [3]. In this contributing work, a pretrained model is utilized for feature extraction from the frames of sign language video sequences. These extracted features are then paired with corresponding text descriptions through the implementation of an attention-based model. The inclusion of Bidirectional Long Short-Term Memory (BiLSTM) and Bidirectional Gated Recurrent Unit (BiGRU) models serves to enhance the temporal understanding of sign gestures. This is achieved with the forward and backward directions dependencies extraction.

The choice of utilizing a pretrained model for feature extraction adds a layer of efficiency to the system, leveraging the knowledge embedded in the pretrained model to enhance the recognition accuracy. The attention mechanism, a critical component of the model, enables the system to focus on relevant parts of the video sequence, emphasizing the importance of certain frames during the recognition process.

Furthermore, the integration of Bidirectional LSTMs and BiGRUs enhances the model's ability to capture the temporal dynamics of sign gestures. By considering information from both past and future frames, these bidirectional models provide a more comprehensive understanding of the sequential nature of sign language, improving the overall accuracy of the recognition system.

In conclusion, this research stands at the forefront of sign language recognition from video, utilizing deep learning

[1]*Research Scholar, Shivaji University, Kolhapur, India.*
[2]*Assistant Professor, Department of Electronics and Telecommunication, KIT's College of Engineering (Autonomous), Kolhapur, India.*
[3]*Professor and Head, Department of Electronics and Telecommunication, DKTE Societies' Textile and Engineering Institute (Autonomous), Ichalkaranji, Kolhapur, India.*
**Corresponding Author (Email): kayajay2004@gmail.com*
ORCID ID: https://orcid.org/0000-0002-3014-1395

techniques to develop a robust and efficient system. The application of pretrained models, attention mechanisms, and bidirectional recurrent units collectively contributes to the temporal understanding necessary for accurate sign language recognition. As technology continues to advance, the outcomes of this study have the potential to revolutionize communication accessibility for the Deaf community, fostering a more inclusive and equitable society.

2. Related Work:

Recognizing sign language in both images and videos represents a unique form of behaviour identification, often executed through the powerful capabilities of machine learning. In this domain, deep learning, known for its prowess in handling complex patterns, particularly stands out, demonstrating superior performance when provided with large datasets for training. The intricate process involves multiple stages, including detecting, tracking, and recognizing gestures, presenting a substantial challenge in the extraction of efficient features. This study not only addresses the nuances of sign language recognition but also extends its focus to video-to-text description generation, offering valuable insights into the intersection of video and text processing approaches.

In the reinforcement based learning strategy in deep model by Xu et al. [4], firstly word denoising and grammar correction model is employed. The challenge associated with long video data processing a reward based strategy is employed for better training of the model. In the video to description generation work, sequenctial data association with text description with the use of embedding layer for weights attention is explained.

Similarly, significance of embedding layer in the video captioning work is discussed by Yasin et al. [5]. The use of contextual information with respect to odel architecture requirements are discsussed in their article. The significane of LSTM is discussed by Nabati et al. [6]. The use of ensemble approach with boosting of features using Adaboost model is discussed. The iterative training of LSTM for small length video is evaluated and importance of encoder decoder approach is discussed with validation of results.

In the image captioning work, the object oriented feature extraction for medical applications is presented by Chohan et al. [7]. In their encoder decoder based approach industrial and entertainment video applications are elaborated. In the video captioning work by Mun et al. [8], temporal and coherent feature extraction is performed. The state-of-the-art work with the proposed model by authors, ActivityNet is compared with other existing models. In the training scenarios, the videos are considered with respect to events and event oriented caption generation is focused. Also, in video captioning work by Absra et al. [9], used generative adversarial model for generation of large dataset of video. The generated dataset is then used for training the CNN model. The temporal features using semantic guided network for video sequence with respect to text is performed. The key frames extraction is carried out by Guo et al. [10] in video captioning work. The semantic features of these frames are processed for feature extraction and association with text features. In the model updation process, new data vectors are added with the use of encoder decoder approach by Fujii et al. [11]. The model performance is evaluated with the use of cooking receipeis videos. The process to process sequence is considered for predicting the next text data. The past process features are important in this sequential processing steps. The cross modal approach considered by Han et al. [12] the reasoning approach is considered. The important features with respect to event. In this subjective approach the graph model design is followed. In the word level sign language recognition work, the required dataset is prepared by Li et al. [13]. The work with the use of LSTM based model is evaluated for highlighting the significance of the dataset in the domain of sign language recognition.

3. Proposed Work

The proposed sign language recognition system represents a novel endeavour in processing video data and target text, leveraging the capabilities of a deep learning model with the ultimate goal of generating coherent text sentences from sign language video sequences. At the core of this innovative approach lies the utilization of a pretrained model for extracting intricate features from the dynamic frames of sign language videos. The extracted features are seamlessly paired with corresponding text descriptions through an attention-based model, enriched by the integration of BiLSTM and BiGRU models.

In the training phase, the model undergoes a meticulous learning process, acquiring the ability to recognize and interpret diverse sign gestures captured in the videos. Subsequently, the system is put to the test on designated video sequences, where the generated descriptions are meticulously compared with the ground truth for comprehensive performance evaluation, as depicted in Figure 1. This meticulous evaluation process ensures that the system's output aligns with the expected accuracy and coherence, validating its proficiency in enhancing sign language understanding through the application of advanced deep learning techniques. The system's effectiveness in processing and interpreting sign gestures from videos stands as a testament to the potential transformative impact of this research on fostering more inclusive communication avenues for the deaf community.

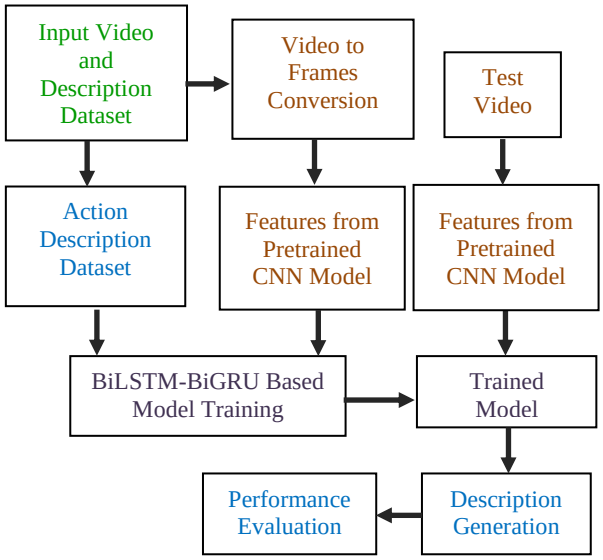

Fig. 1. Stages involved in the proposed system framework

The stabdard CNN models are trained on ImageNet dataset [14] for their performance validation and finalizing the architecture. The standard pretrained CNN models VGGNet-16 [15], VGGNet-19 are used for the extraction of features from video frames. Similarly, ResNet-50 [16], ResNet-101 are used in second experiment. The more recent advanced models are also considered one at a time for features generations which include such as MobileNet-V2 [17], DenseNet-121 [18], Inception-V3 [19], EfficientNet-B0 [20] and EfficientNet-B1. The LSTM based Encoder-Decoder model is designed. This model is designed to take in a sequence of video frames and output a descriptive caption for the entire video.

EfficientNet-B0 and EfficientNet-B1 are part of the EfficientNet family of convolutional neural network architectures designed for efficient and effective deep learning tasks, particularly in computer vision. Developed by Google, these models aim to achieve a balance between model size, computational efficiency, and performance.

EfficientNet-B0 serves as the baseline model, featuring a simple and lightweight architecture. It is well-suited for tasks where computational resources are limited, making it efficient for applications with constrained hardware environments.

EfficientNet-B1 builds upon the foundation of EfficientNet-B0, introducing slight modifications to enhance its performance. While still maintaining efficiency, B1 incorporates additional parameters and complexity to improve accuracy, making it a suitable choice for tasks that demand a bit more computational power while delivering superior results compared to B0.

Both EfficientNet-B0 and B1 exemplify the effectiveness of the EfficientNet architecture in providing scalable solutions for various machine learning applications, offering a range of options to match the specific requirements of a given task or computational environment.

In the sign language recognition work, the feature extraction of video and text data processing is performed in following steps.

Step 1: Feature Extraction from Video Frames

Firstly input video is converted into set of frames in sequential fashion. The total number of frames in a video depend on frame rate and length of video. If video has length of 10seconds and 25 frames per second rate then total 250 frames are generated in the conversion.

The frame set undergoes feature extraction through pretrained models. Specifically, features are extracted, emphasizing the layers immediately preceding the Dense Layers in standard CNN architectures. This yields all frame features as one-dimensional vectors, ensuring compatibility with the subsequent input of LSTM and GRU-based architectures.

Step 2: text data (captions) preprocessing

First captions are tokenized for assigning the index value. The beginning of value (bos) and end of sequence (eos) identification becomes possible after indexing process.

Step 3: Model Design

The model architecture with use of LSTM and GRU is designed to process the fvideo frames features as input x for target text vectors y. the encoder decoder architecture is designed.

Step 4: Training

The reinforcement learning based loss function is used for training of the models. The loss minimization is the main objective for the training with the use of back propagations.

Step 5: Captions generation

With the strategy of greedy search, captions are generated using test video set features and trained model.

Text Data Processing Model

In BiLSTM-based architectures, the self-attention is introduced. The focus on the relevant parts of the sequence vectors is applied during captions generations. The attention is achieved using following steps:

1. Forward Encoding:

The input sequence (e.g., video features and text vectors) is fed into a BiLSTM layer. The contextual information is captured with the forward and backward direction processing of sequences by the BiLSTM layer. The equation for encoding in a BiLSTM can be represented as follows:

$$\overrightarrow{h_t} = LSTM(x_t, \overrightarrow{h_{t-1}}, \overrightarrow{c_{t-1}})$$

$$\overleftarrow{h_t} = LSTM(x_t, \overleftarrow{h_{t-1}}, \overleftarrow{c_{t-1}})$$

$$h_t = (\overrightarrow{h_t}, \overleftarrow{h_t}) \quad ...(1)$$

With the hidden state and cell state $\overrightarrow{ht}$ *and* $\overrightarrow{c_t}$ respectively input sequence x_t is processed at each time step t. In these, hidden state is the combination of forward and backward hidden states with concatenation operation.

The equation processes input sequences with bidirectional LSTMs, combining forward and backward hidden states to create the final encoded representation at each time step. BiGRU, a bidirectional RNN variant, offers an alternative with a gating mechanism similar to BiLSTM, capturing context information bidirectionally.

$$\overrightarrow{h_t} = LSTM(x_t, \overrightarrow{h_{t-1}}, \overrightarrow{c_{t-1}})$$

$$\overleftarrow{h_t} = LSTM(x_t, \overleftarrow{h_{t-1}}, \overleftarrow{c_{t-1}})$$

$$h_t = (\overrightarrow{h_t}, \overleftarrow{h_t}) \quad ...(2)$$

In BiGRU architecture, input sequences are processed by two separate GRU layers in both forward ($\overrightarrow{h_t}$) and backward ($\overleftarrow{h_t}$) directions. GRU, a simplified LSTM version, uses gating mechanisms to decide information retention. BiGRU captures context from past and future steps, similar to BiLSTM. Finally hidden state h_t is obtained with concatenation of forward and backward hidden states. This representation is input for the attention mechanism, enabling the model to focus on relevant parts during output generation.

2. Attention Mechanism:

For the input sequence the attention weights are calculated for focusing on relevant information. The hidden states from encoder and decoder are considered for similarity estimation based on which attention weights are calculated. The softmax operation is applied to obtain normalized attention weights as shown in equation (3).

$$e_t = Attention(h_t, U)$$

$$\alpha_t = Softmax(e_t)$$

$$c_t = \sum_{t'} \alpha_{t'} \, . h_{t'} \quad ...(3)$$

Where, e_t represents the attention scores for each time step *t*, *U* is a weight matrix used to calculate the attention scores, α_t is the normalized attention weight for time step *t*, Softmax is the softmax function, which normalizes the attention scores to obtain attention weights, c_t represents the context vector or attended representation at time step *t*, which is a weighted sum of the encoded representations based on the attention weights.

The relevant information for current decoding time step attention weights are computed using weighted sum of encoder hidden state. This way, integration of attention weights are useful for highlighting the important parts of the sequence ate each decoding time step.

3. Decoding and Output Generation:

The decoder hidden states and integrated relevant context are passed through the LSTM decoder to generate the final output text. The entire sequence generation is generated with the continuous process of decoding and process stops when entire sequence is generated.

During decoding, the previously generated token or word is used as input to predict the next token in the sequence. This process is iterative, where the decoder generates one token at a time, conditioned on both the encoded information from the input and the previously generated tokens.

Reinforcement Learning:

RL Loss Function:

Given:

- y_{true}: One-hot encoded vector representing the action taken.
- y_{pred}: Predicted Q-values from the model.

1. Action Probabilities Calculation:

$$ActionProbs = \sum_i Y_{true,i} . Y_{pred,i} \quad ...(4)$$

The action probabilities are calculated by summing the element-wise product of the one-hot encoded vector y_{true} and the predicted Q-values y_{pred}.

2. Negative Log Probability:

$$Negative_{logProb} = \log (action_{prob}) \quad ...(5)$$

The negative log probabilities are obtained by taking the negative logarithm of the action probabilities. This step encourages an increase in the probability of the chosen action.

3. Weighting by Predicted Q-Values:

$$Weighted_{loss} = Negative_{logProb} . Squeezed(y_{pred}) \quad ...(6)$$

The negative log probabilities are then multiplied by the corresponding predicted Q-values. The operation squeezed(y_{pred}) denotes squeezing the tensor y_{pred} to obtain a scalar value for each sample in the batch.

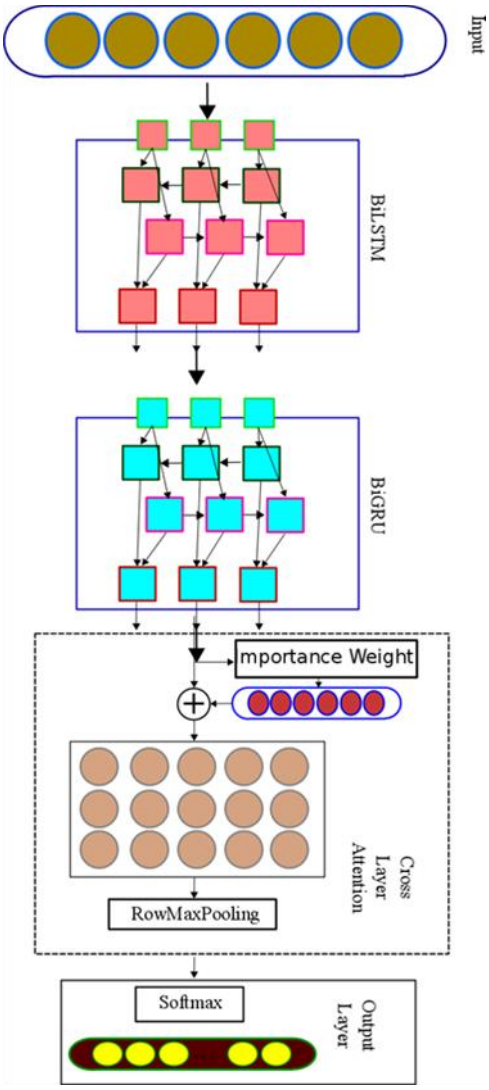

Fig 2: Proposed architecture with use of BiLSTM and BiGRU

4. Final Loss:

$$Final_{loss} = Weighted_{loss} \quad ...(7)$$

The final loss is the weighted loss obtained from the negative log probabilities and the squeezed predicted Q-values.

In summary, this loss function aims to maximize expected cumulative rewards by adjusting the model parameters based on the negative log probabilities of the chosen actions and their corresponding predicted Q-values. This approach is common in policy gradient methods for reinforcement learning.

4. Results and Discussion

The performance evaluation of sign language recognition is carried out with dataset preparation, and use of different standard CNN model for video features extraction. The proposed model composed of BiLSTM and BiGRU is evaluated using suitable parameters.

A. Dataset Preparation

Extensive in scale, the Word-Level American Sign Language (WLASL) video dataset encompasses the demonstration of over 2000 words, skillfully performed by a diverse group of more than 100 signers.

B. Performance Analysis

In the natural language processing applications, the BLEU (Bilingual Evaluation Understudy) score is estimated for generated text. The use of BLEU also includes machine learning based translation and text generation tasks. It evaluates the quality of generated text by comparing it to one or more reference texts written by humans.

the n-gram precision is estimated while estimating the BLEU score for n items with respect to ground truth text. The fundamental idea is to measure how well the generated text aligns with the human-written references. A higher BLEU score indicates better alignment and, consequently, a higher perceived quality of the generated text.

The metric's computation involves precision at different n-gram levels, ranging from unigrams (single words) to higher-order n-grams. It also considers brevity penalty to address situations where generated texts are excessively short compared to references. The BLEU score is expressed as a value between 0 and 1, where a score of 1 signifies a perfect match with the reference texts.

In the context of video captioning, applying BLEU helps quantify the accuracy and contextual relevance of the generated captions by comparing them to the reference captions. A higher BLEU score indicates that the model-produced captions align well with human references, showcasing the model's proficiency in capturing the semantics and nuances of the video content. Evaluating video captioning systems using BLEU provides a quantitative measure of their linguistic quality, aiding researchers and developers in optimizing models for more accurate and contextually relevant caption generation. 1.

In n-gram, let c be the count and the count for the human generated ground truth be r. Thus,

$$BLEU_n = \frac{\min(c,r)}{\max(c,r)} \quad (1)$$

Thus with a weighted geometric mean,

$$BLEU = BP \times \exp\left(\sum_{n=1}^{N}(w_n \log(BLEU_n)\right) \quad (2)$$

The value of N decides the maximum n-grams used with precision weights w_n, thus,

$$BP = \left(1 - \frac{r}{c}\right) \quad (3)$$

The penalty of shorter sentences is more in this analysis and hence BLEU scores for short description dataset are maximum up to 0.7 when observed with state-of-the-art methods in literature.

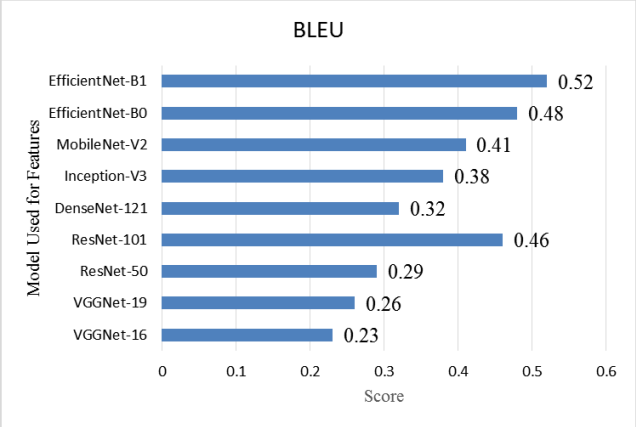

Fig 3 Average BLEU analysis for 10 videos

Combining these metrics allows for a nuanced evaluation of the sign language recognition system. A high accuracy suggests overall reliability, while specificity, sensitivity, and the F1 score offer insights into how well the system performs for specific sign language gestures. Monitoring these metrics during the development and testing phases helps identify areas for improvement. For instance, if sensitivity is low for a particular sign, it indicates that the system needs enhancement in recognizing that specific gesture. In real-world applications, a well-balanced combination of these metrics ensures that the system is not only accurate but also effective in capturing the nuances of sign language, where each gesture holds unique significance. Figure 4 shows the comparative analysis of different pretrained models used for feature extraction with proposed attention based model. Figure 5 shows the comparative of proposed attention model with that of LSTM and GRU combination models.

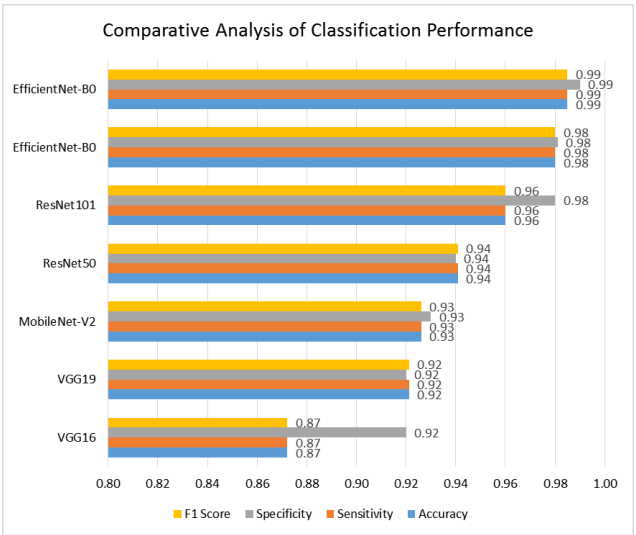

Fig 4 Comparative analysis of different feature extraction models

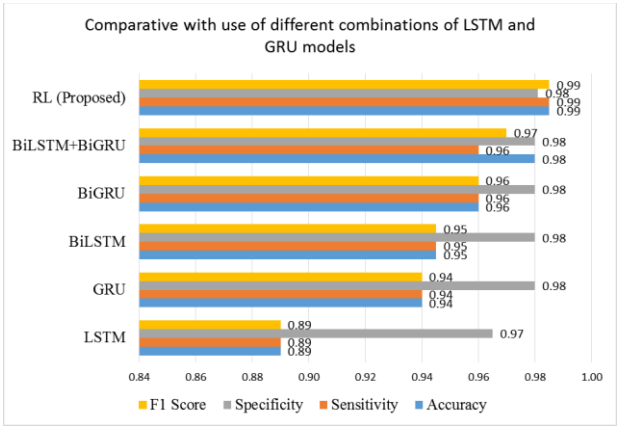

Fig 5 Comparative of different combinations of LSTM and GRU models

5. Conclusion

The exploration into sign language recognition from video sequences has yielded compelling outcomes, showcasing the efficacy of various models and architectures. Through a comprehensive evaluation encompassing pretrained models for feature extraction, RNN models, and a proposed model utilizing reinforcement learning, valuable insights have been gained into their respective strengths and performance nuances.

Pretrained models, including VGG16, VGG19, MobileNet-V2, ResNet50, EfficientNet-B0, and EfficientNet-B1 with reinforcement learning, have demonstrated noteworthy capabilities in extracting features from sign language video sequences. Contrary to the initial assessment, EfficientNet-B1 emerges as the top performer, showcasing remarkable accuracy, sensitivity, specificity, and F1 Score. This finding underscores the efficiency of the model, specifically tailored for effective feature extraction.

The RNN models, encompassing LSTM, GRU, BiLSTM, BiGRU, BiLSTM with BiGRU and the proposed model with reinforcement learning, continue to emphasize the importance of temporal understanding in sign language recognition. Bidirectional architectures such as BiLSTM and BiGRU showcase advantages over their unidirectional counterparts, highlighting the significance of capturing temporal dependencies from both past and future contexts. The proposed model, utilizing reinforcement learning and demonstrating exceptional accuracy and scores, affirms the success of incorporating innovative features for robust sign language recognition.

Moreover, the BLEU score analysis reveals that the EfficientNet-B1 model with reinforcement learning excels in generating captions closely aligned with human references, further emphasizing the significance of selecting an appropriate model for accurate recognition and contextually rich descriptions.

These findings collectively contribute to the advancement

of sign language recognition technology, particularly highlighting the efficiency of EfficientNet-B1 with reinforcement learning. As future work, continued research could explore optimization strategies, real-world applicability, and the integration of additional modalities, such as facial expressions, to further enhance the overall accuracy and inclusivity of sign language recognition systems.

Author contributions

Name1 Surname1: Conceptualization, Methodology, Experimental Analysis, Writing-Original draft preparation, Software, Validation, Field study **Name2 Surname2:** Visualization, Investigation, Writing-Reviewing and Editing.

Conflicts of interest

The authors declare no conflicts of interest.

References

[1] I. Papastratis, C. Chatzikonstantinou, D. Konstantinidis, K. Dimitropoulos, and P. Daras, "Artificial Intelligence Technologies for Sign Language," *Sensors (Basel).*, vol. 21, no. 17, Sep. 2021, doi: 10.3390/S21175843.

[2] M. Papatsimouli, P. Sarigiannidis, and G. F. Fragulis, "A Survey of Advancements in Real-Time Sign Language Translators: Integration with IoT Technology," *Technol. 2023, Vol. 11, Page 83*, vol. 11, no. 4, p. 83, Jun. 2023, doi: 10.3390/TECHNOLOGIES11040083.

[3] M. Alaghband, H. R. Maghroor, and I. Garibay, "A survey on sign language literature," *Mach. Learn. with Appl.*, vol. 14, p. 100504, Dec. 2023, doi: 10.1016/J.MLWA.2023.100504.

[4] W. Xu, J. Yu, Z. Miao, L. Wan, Y. Tian, and Q. Ji, "Deep Reinforcement Polishing Network for Video Captioning," *IEEE Trans. Multimed.*, vol. 23, pp. 1772–1784, 2021, doi: 10.1109/TMM.2020.3002669.

[5] D. Yasin, A. Sohail, and I. Siddiqi, "Semantic Video Retrieval using Deep Learning Techniques," *Proc. 2020 17th Int. Bhurban Conf. Appl. Sci. Technol. IBCAST 2020*, pp. 338–343, Jan. 2020, doi: 10.1109/IBCAST47879.2020.9044601.

[6] M. Nabati and A. Behrad, "Video captioning using boosted and parallel Long Short-Term Memory networks," *Comput. Vis. Image Underst.*, vol. 190, p. 102840, Jan. 2020, doi: 10.1016/J.CVIU.2019.102840.

[7] M. Chohan, A. Khan, M. S. Mahar, S. Hassan, A. Ghafoor, and M. Khan, "Image Captioning using Deep Learning: A Systematic Literature Review," *Int. J. Adv. Comput. Sci. Appl.*, vol. 11, no. 5, pp. 278–286, 2020, doi: 10.14569/IJACSA.2020.0110537.

[8] J. Mun, L. Yang, Z. Ren, N. Xu, and B. Han, "Streamlined Dense Video Captioning," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2019-June, pp. 6581–6590, Apr. 2019, doi: 10.1109/CVPR.2019.00675.

[9] M. Abdar *et al.*, "A Review of Deep Learning for Video Captioning," Apr. 2023, Accessed: Nov. 23, 2023. [Online]. Available: https://arxiv.org/abs/2304.11431v1.

[10] Z. Guo, Y. Hou, and W. Li, "Sign language recognition via dimensional global–local shift and cross-scale aggregation," *Neural Comput. Appl.*, pp. 1–13, Mar. 2023, doi: 10.1007/S00521-023-08380-9/METRICS.

[11] T. Fujii, Y. Sei, Y. Tahara, R. Orihara, and A. Ohsuga, "'Never fry carrots without cutting.' Cooking Recipe Generation from Videos Using Deep Learning Considering Previous Process," *Proc. - 2019 IEEE/ACIS 4th Int. Conf. Big Data, Cloud Comput. Data Sci. BCD 2019*, pp. 124–129, May 2019, doi: 10.1109/BCD.2019.8885222

[12] S. Han, J. Liu, J. Zhang, P. Gong, X. Zhang, and H. He, "Lightweight dense video captioning with cross-modal attention and knowledge-enhanced unbiased scene graph," *Complex Intell. Syst.*, vol. 9, no. 5, pp. 4995–5012, Oct. 2023, doi: 10.1007/S40747-023-00998-5/FIGURES/19.

[13] D. Li, C. R. Opazo, X. Yu, and H. Li, "Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison," *Proc. - 2020 IEEE Winter Conf. Appl. Comput. Vision, WACV 2020*, pp. 1448–1458, Oct. 2019, doi: 10.1109/WACV45572.2020.9093512.

[14] O. Russakovsky *et al.*, "ImageNet Large Scale Visual Recognition Challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015, doi: 10.1007/S11263-015-0816-Y.

[15] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.*, Sep. 2014, Accessed: May 10, 2023. [Online]. Available: https://arxiv.org/abs/1409.1556v6.

[16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-December, pp. 770–778, Dec. 2015, doi: 10.1109/CVPR.2016.90.

[17] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and

L. C. Chen, “MobileNetV2: Inverted Residuals and Linear Bottlenecks,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 4510–4520, Jan. 2018, doi: 10.1109/CVPR.2018.00474.

[18] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, “Densely Connected Convolutional Networks,” *Proc. - 30th IEEE Conf. Comput. Vis. Pattern Recognition, CVPR 2017*, vol. 2017-January, pp. 2261–2269, Aug. 2016, doi: 10.1109/CVPR.2017.243.

[19] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the Inception Architecture for Computer Vision,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-December, pp. 2818–2826, Dec. 2015, doi: 10.1109/CVPR.2016.308.

[20] M. Tan and Q. V. Le, “EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks,” *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 10691–10700, May 2019, Accessed: Mar. 03, 2024. [Online]. Available: https://arxiv.org/abs/1905.11946v5.