Real-Time Sign Language Fingerspelling Recognition System Using 2D Deep CNN with Two-Stream Feature Extraction Approach

Aziza Zhidebayeva[1], Gulira Nurmukhanbetova[2], Sapargali Aldeshov[3],
Kamshat Zhamalova[4], Satmyrza Mamikov[5], Nursaule Torebay[6]
University of Friendship of People's Academician A. Kuatbekov, Shymkent, Kazakhsatan[1, 5]
South Kazakhstan Pedagogical University named after Ozbekali Zhanibekov, Shymkent, Kazakhstan[2, 3, 4]
Miras University, Shymkent, Kazakhstan[6]

***Abstract*—This research paper introduces a novel sign language recognition system developed using advanced deep learning (DL) techniques aimed at enhancing communication capabilities between deaf and hearing individuals. The system leverages a convolutional neural network (CNN) architecture, optimized for the real-time interpretation of dynamic hand gestures that constitute sign language. A comprehensive dataset was employed to train and validate the model, encompassing a diverse range of gestures across different environmental settings. Comparative analysis revealed that the deep learning-based model significantly outperforms traditional machine learning techniques in terms of recognition accuracy, particularly with the increase in the volume of training data. This was illustrated through various performance metrics, including a detailed confusion matrix and Levenshtein distance measurements, highlighting the system's efficacy in accurately identifying complex gestures. Real-time application tests further demonstrated the model's robustness and adaptability to varying lighting conditions and backgrounds, essential for practical deployment. Key challenges identified include the need for broader linguistic diversity in training datasets and enhanced model sensitivity to subtle gestural distinctions. The paper concludes with suggestions for future research directions, emphasizing algorithm optimization, data diversification, and user-centric design improvements to foster wider adoption and usability. This study underscores the potential of deep learning technologies to revolutionize assistive communication tools, making them more accessible and effective for the deaf community.**

***Keywords*—Deep learning; sign language recognition; convolutional neural networks; real-time processing; gesture recognition; machine learning; accessibility technology**

I. Introduction

Fingerspelling is a critical component of sign languages, used primarily for spelling out words that do not have established signs, such as proper nouns and technical terms. This aspect of sign language communication has drawn significant attention in the realm of automated recognition systems, driven by the potential to bridge communication gaps between deaf and hearing communities. The development of a real-time fingerspelling recognition system using two-dimensional deep convolutional neural networks (2D Deep CNNs) represents a significant leap toward inclusive communication technologies. This paper aims to develop a real-time recognition system that utilizes advanced deep learning methodologies to accurately recognize fingerspelling from video inputs in sign language.

The importance of addressing the nuances in sign language through technological solutions cannot be overstated. Sign languages, unlike spoken languages, utilize manual communication and body language to convey meaning, incorporating a complex combination of hand shapes, orientations, movements, and facial expressions [1]. Fingerspelling is an integral component, especially in educational settings and daily communication, where specific terminology and names need clear articulation [2]. However, the manual and non-manual components of sign language pose unique challenges in automatic recognition, which must be addressed to achieve high accuracy and real-time performance [3].

Recent advancements in deep learning have shown promising results in image and video recognition tasks, which are pivotal in interpreting dynamic and complex gestures in sign language [4]. Particularly, the application of 2D Deep CNNs has emerged as a potent approach due to their ability to extract spatial hierarchies of features from visual data [5]. These networks have been effectively employed in various domains, including facial recognition and autonomous driving, underscoring their versatility and robustness in handling complex visual data [6].

The concept of using a two-stream feature extraction approach in our system is inspired by the successes seen in action recognition in videos, where separate streams are used to capture spatial and temporal features [7]. In the context of fingerspelling, one stream processes static images to recognize hand shapes, while the other captures the motion between frames to understand the dynamics of hand movements [8]. This dual approach is designed to enhance the system's ability to discern subtle differences in fingerspelling gestures, which are often rapid and involve minimal but significant movements.

However, the challenge in developing such systems is not only technical but also linguistic. Sign languages are not universal; thus, a model trained on one sign language may not be applicable to another [9]. This diversity necessitates adaptable models that can learn from limited data and

generalize well across different languages and even dialects within the same language [10]. Additionally, the environmental variability in video data, such as background complexity, lighting conditions, and camera angles, also affects the performance of recognition systems [11].

Another critical aspect is the real-time capability of the system, which is essential for practical applications. For users to adopt such a technology effectively, the recognition process must be fast enough to occur in natural conversation time without significant delays [12]. Achieving this requires not only robust model architecture but also optimized computation strategies to process video frames swiftly [13].

In this study, we propose a system architecture that incorporates a streamlined 2D Deep CNN with a two-stream feature extraction strategy tailored for real-time application. Previous research has indicated the feasibility of real-time processing using deep learning models, particularly those optimized for mobile and embedded systems [14]. By integrating such models with a two-stream approach, we aim to achieve a balance between accuracy and speed, making the system practical for everyday use [15].

Furthermore, the development of such systems also opens avenues for enhanced educational tools, accessible services, and improved autonomy for the deaf and hard-of-hearing community [16]. As technology progresses, the integration of such specialized communication tools could profoundly impact social inclusion and equality.

This paper explores the technical development of the proposed recognition system, evaluates its performance across various metrics, and discusses its potential applications and implications for the future of communication technologies in the context of sign language. By pushing the boundaries of what is possible in automated fingerspelling recognition, we aim to contribute to a more inclusive and accessible technological landscape.

II. Related Work

The development of fingerspelling recognition systems using computer vision and machine learning technologies has garnered considerable attention in the academic community. This section reviews existing literature related to the application of deep learning techniques for sign language recognition, with a specific focus on fingerspelling, feature extraction methodologies, real-time processing capabilities, and the challenges posed by diverse sign languages.

A. 2D and 3D Convolutional Neural Networks in Sign Language Recognition

Recent studies have extensively employed convolutional neural networks (CNNs) for the task of sign language recognition. The application of 2D CNNs has proven effective in recognizing static sign language images, capturing spatial features that distinguish various hand signs [17]. However, the dynamic nature of sign language, particularly in fingerspelling, requires understanding temporal sequences, for which 3D CNNs are better suited. These networks extend the capability of 2D CNNs by adding time as a third dimension, allowing them to capture motion across frames effectively [18]. A notable study demonstrated the superiority of 3D CNNs over their 2D counterparts in recognizing continuous sign language gestures, attributing improvements to the network's ability to process temporal information [19]. Nevertheless, the computational demand of 3D CNNs remains a significant challenge, particularly for real-time applications [20].

B. Feature Extraction Techniques for Enhanced Gesture Recognition

The effectiveness of a recognition system largely depends on the robustness of its feature extraction process. In this context, two-stream CNN architectures have shown promising results by separately processing spatial and temporal features, thus providing a more comprehensive analysis of video data [21]. One stream typically processes individual frames to capture static features like hand shapes and positions, while the other analyzes motion between frames to capture dynamic movements [22]. Such approaches have been applied successfully in other fields of action recognition and have gradually been adapted for sign language recognition [23]. Hybrid models that combine CNNs with recurrent neural networks (RNNs) have also been explored, with RNNs processing the temporal sequences of features extracted by CNNs, thereby enhancing the recognition of gestures over time [24]. Moreover, attention mechanisms have been integrated to focus the model on relevant features of the hand, significantly improving accuracy by reducing the influence of background noise and other irrelevant signals [25].

C. Real-Time Processing for Sign Language Recognition Systems

Achieving real-time processing capabilities in sign language recognition systems is crucial for their practical application. The latency in processing and recognizing sign language must be minimized to facilitate fluid communication between deaf and hearing individuals. Several studies have focused on optimizing CNN architectures to reduce computational loads without compromising accuracy [26]. Techniques such as model pruning, quantization, and the use of efficient network architectures like MobileNets have been proposed as solutions to achieve faster processing times [27]. Furthermore, edge computing has emerged as a viable approach, where processing is done on local devices rather than relying on cloud-based systems, thereby reducing response times significantly [28]. These advancements have paved the way for the development of more responsive and efficient real-time sign language recognition systems [29].

D. Challenges in Multilingual Sign Language Recognition

Sign language recognition is further complicated by the variation in sign languages across different regions and cultures. Each sign language has its own set of rules and nuances, which means that a system trained on one language may not perform well on another [30]. The scarcity of annotated datasets for many sign languages poses a significant barrier to training robust models [31]. Studies have attempted to address these challenges by using transfer learning, where a model trained on one language is adapted to another with minimal additional training [32]. Another approach is the use of synthetic data generation to augment existing datasets, thereby providing more comprehensive training material [33]. These

methods have shown some success, but the variability in performance across languages remains a concern [34].

The literature reviewed highlights significant advancements in the field of sign language recognition, particularly in applying deep learning techniques for fingerspelling recognition. While 2D and 3D CNNs offer robust frameworks for feature extraction, their integration with two-stream architectures and RNNs presents a promising path toward more accurate and efficient recognition systems. Real-time processing remains a critical area for ongoing research, with current solutions pointing towards optimized CNN architectures and edge computing. However, the multilingual and multicultural nature of sign languages continues to pose significant challenges, necessitating further research into adaptive and scalable models that can handle the diversity of sign languages globally.

III. Materials and Methods

This section is critical as it outlines the systematic steps taken to ensure the reliability and validity of the results obtained. It serves to offer transparency, allowing other researchers to replicate the study or build upon its findings. Within this section, we detail the specific datasets used, the data preprocessing techniques employed, the architectural design of the machine learning models, and the criteria for evaluating their performance. By providing a clear and thorough exposition of these elements, we aim to facilitate a deeper understanding of the research process and its foundational components.

A. Sign Language Alphabets

Sign language alphabets serve as fundamental building blocks for communication within deaf communities, providing a means to spell out words and names for which specific signs may not exist. Among the various sign languages utilized globally, American Sign Language (ASL) [35] and Indian Sign Language (ISL) [36] represent two distinct systems, each with its unique set of alphabetic representations. This section delves into the alphabetic systems of ASL and ISL, illustrating their characteristics and the cultural nuances that influence their formation and usage.

1) American Sign Language (ASL) Alphabet: American Sign Language (ASL) is one of the most widely used sign languages in the world, particularly prevalent in the United States and parts of Canada. The ASL alphabet, as depicted in Fig. 1, consists of a series of hand configurations used to represent the 26 letters of the English alphabet. Each letter is formed using one hand, which is a notable characteristic that distinguishes ASL from some other sign languages that might use two hands for certain letters. The ASL fingerspelling system is crucial for expressing proper nouns, technical terms, and any other words for which there is no established sign, thus playing a vital role in daily communication as well as educational settings [37].

The ASL alphabet's design emphasizes clarity and simplicity, allowing for quick and straightforward communication. The letters are generally formed in front of the signer at chest level, ensuring visibility and ease of understanding. For instance, the letters 'A' through 'Z' involve distinct positions and shapes of the fingers, with minimal movement, making them relatively easy to learn for beginners and highly functional for fluent users in rapid communication.

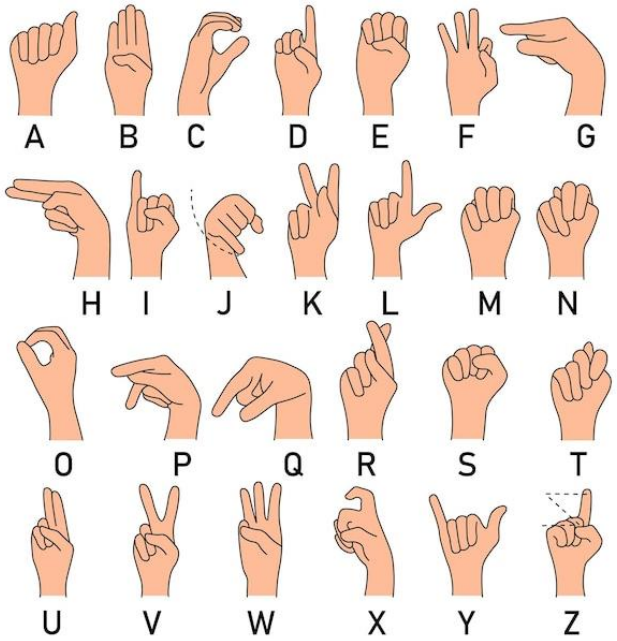

Fig. 1. ASL Alphabet.

2) Indian Sign Language (ISL) Alphabet: In contrast, Indian Sign Language (ISL) caters to the diverse linguistic landscape of India and incorporates elements that reflect the cultural and regional diversity of the country. The ISL alphabet, shown in Fig. 2, is utilized across various educational and social settings in India, providing a means for the deaf community to partake in both public and private discourse. Unlike ASL, the ISL fingerspelling system often employs two hands to represent certain letters, which can be seen as an adaptation to the linguistic structures and phonetic complexities of the multiple languages spoken in India [38].

Each letter in the ISL alphabet is represented by a unique combination of hand shapes, positions, and movements. These elements are designed to be visually distinct from one another to minimize confusion and ensure effective communication. For instance, the letters of the ISL alphabet are depicted with both static and dynamic gestures, which involve more interaction between the two hands compared to the mostly static nature of ASL fingerspelling [39]. This characteristic of ISL may stem from the gestural nuances found in the native languages of India, which often emphasize expressive hand movements and gestures in daily communication.

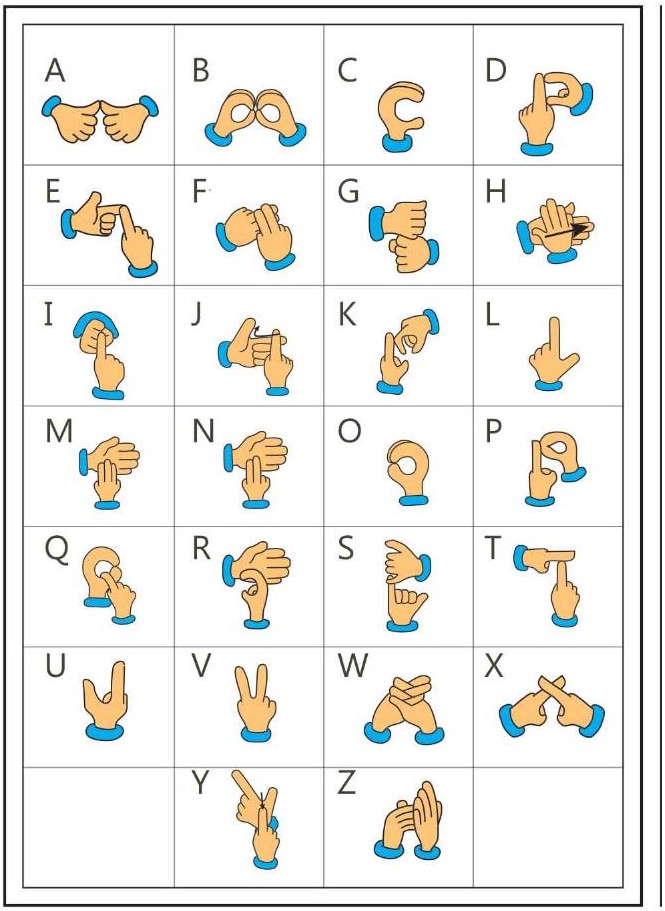

Fig. 2. ISL Alphabet.

The alphabets of ASL and ISL highlight the adaptability and diversity of sign languages in accommodating the linguistic needs of different cultural contexts [40-42]. While ASL employs a one-handed system for simplicity and speed, ISL uses a two-handed approach, possibly reflecting the complex phonetic systems of India's numerous spoken languages. Understanding these differences is crucial for educators, linguists, and technologists who develop communication tools and educational materials for the deaf community. The study and comparison of such systems not only enhance our understanding of linguistic diversity but also promote more inclusive and effective communication tools tailored to the unique needs of each sign language community.

B. Data Structure and Sample Description

Fig. 3 illustrates the data structure used in the research project for training the sign language recognition model. The data organization is key to understanding the relational dynamics between different datasets, which include train.csv and 5414471.parquet. The diagram effectively shows how these files are interlinked and utilized to train the deep learning model, highlighting the integration of participant information, sequence IDs, and feature vectors extracted from video frames.

1) Train.csv**:** The train.csv file acts as a central index, containing essential metadata for each training sample. This file includes several columns:

- path: Specifies the location of the parquet file containing the detailed sequence data.
- file_id: A unique identifier for the parquet file.
- sequence_id: A unique identifier that links the train.csv entries with specific sequences in the 5414471.parquet file.
- participant_id: Identifies the participant from whom the data was collected, facilitating analysis on a per-subject basis if required.
- phrase: Represents the specific phrase or words being signed in the sequence, which is crucial for supervised learning where the model learns to associate specific gestures with their corresponding linguistic outputs.

2) 5414471.parquet: The 5414471.parquet file contains detailed frame-by-frame data extracted from video sequences of participants performing sign language. Each row in this file corresponds to a single frame from a video sequence, and is linked back to the train.csv via the sequence_id. The columns in this file include:

- sequence_id: Matches the sequence_id in train.csv, establishing a relational link.
- frame: The frame number within a particular video sequence, which is critical for analyzing the temporal progression of gestures.
- x_face_0, x_face_1, ...: These columns represent extracted feature vectors associated with each frame. The features might include positional data of different facial landmarks or other relevant metrics that are used as input for the deep learning model.

The diagram in Fig. 3 demonstrates the workflow from raw video data extraction through to the feature extraction process, ending with the data being formatted into a machine-readable structure for model training. This structure supports the development of a robust model by providing a comprehensive dataset that includes both the static context of the sign language phrases and dynamic motion information encapsulated in the sequence of frames. This detailed and methodical data organization ensures that the machine learning model can learn from a rich dataset that mimics the complexities of real-world sign language usage.

C. Proposed Model Architecture

The proposed model architecture for sign language recognition, illustrated in Fig. 4, is designed to process sequential image data through a series of convolutional and fully connected layers. This architecture harnesses the power of deep learning to effectively capture both spatial and temporal features critical for understanding dynamic sign language gestures. Below, we describe each component of the model as depicted in the figure, and detail the operations performed at each stage.

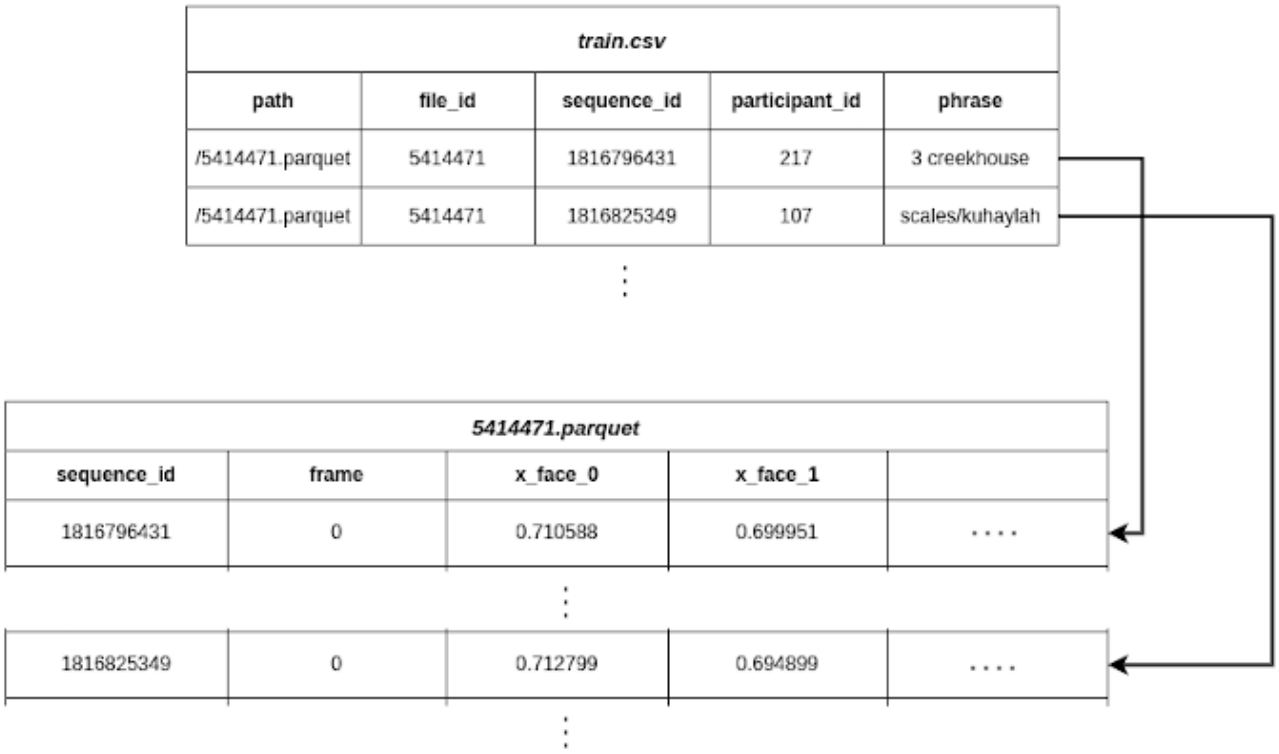

Fig. 3. Example of training sample.

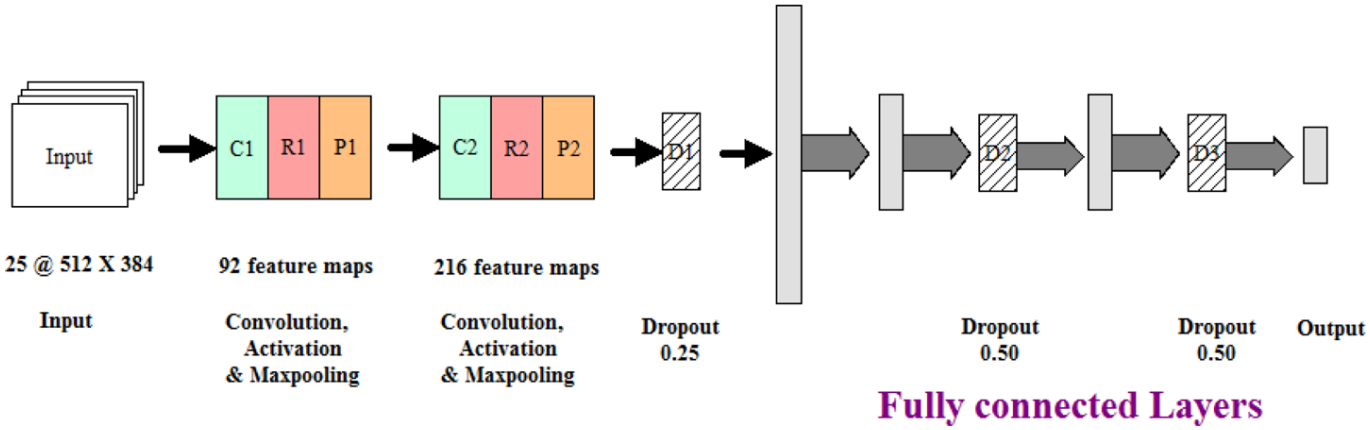

Fig. 4. The proposed model.

1) Input Layer: The input layer accepts sequential image data with dimensions 25×512×384, where 25 represents the number of frames in the sequence, and 512×384 are the pixel dimensions of each frame. This three-dimensional input is essential for preserving the spatial and temporal information present in video data.

Convolutional Layers (C1, C2). The first part of the model consists of two convolutional layers labeled as C1 and C2. These layers are responsible for extracting high-level features from the input images through the application of filters.

C1 Layer: Applies 92 filters to the input images, generating 92 feature maps. The convolution operation can be represented by the equation:

$$F_{ij}^{(k)} = \sigma\left(\sum_{m,n} I_{i+m,j+n} \cdot K_{mn}^{(k)} + b_k\right) \quad (1)$$

Here, $F_{ij}^{(k)}$ is the feature map for the k-th filter at position (i, j), σ is the nonlinear activation function (typically ReLU), I is the input matrix, $K^{(k)}$ is the k-th filter matrix, and b_k is the bias term.

C2 Layer: Further refines the features extracted by the first layer by applying additional 216 filters, thus producing 216

feature maps. This layer helps in capturing more complex features that are vital for accurate sign language recognition.

2) Activation and Pooling Layers (R1, P1, R2, P2): After each convolutional layer, the model applies an activation function followed by a max pooling operation:

Activation (ReLU): Enhances non-linearity in the model by applying the ReLU activation function, which helps in handling non-linear features efficiently [43].

Max Pooling: Reduces the spatial dimensions of the feature maps while retaining the most significant information. This operation is critical for reducing the computational complexity and improving the robustness of the model against small variations in the input data.

3) Dropout layers: Following the convolutional blocks, two dropout layers are included to prevent overfitting [44]:

Dropout 0.25: Applied after the first convolutional block, this layer randomly sets a fraction of input units to 0 at each update during training time, which helps in making the model more generalized.

Dropout 0.50: A higher dropout rate is used after the second convolutional block to further regularize the model, particularly important due to the increased complexity from more feature maps.

4) Fully connected layers: The model transitions from convolutional layers to fully connected (dense) layers, which are essential for making predictions [45]. The dense layers integrate the learned features from previous layers to form the final output. The sequence of fully connected layers ends in a softmax or sigmoid output layer (depending on the number of classes), which provides the probabilities of each sign language gesture.

5) Output Layer: The final layer of the model uses a softmax activation function to classify the input image sequence into one of the possible sign language gestures. The softmax function is given by [46]:

$$P(y = j \mid x) = \frac{e^{z_j}}{\sum_{k=1}^{K} e^{z_k}} \tag{2}$$

Where $P(y = j \mid x)$ is the probability that the input x belongs to class j, z_j is the input to the softmax function from the final fully connected layer for class j, and K is the total number of classes.

The architecture proposed in Fig. 4 is designed to be robust, efficient, and capable of handling the complexities associated with recognizing sign language from video sequences. The combination of convolutional layers for feature extraction and fully connected layers for classification forms a powerful model that is well-suited for the real-time interpretation of sign language.

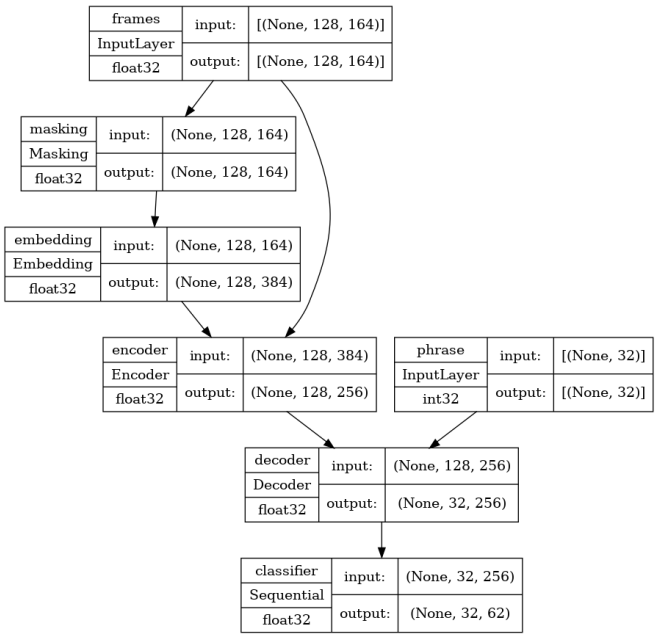

Fig. 5. Flowchart of the proposed model.

Fig. 5 presents a flowchart of a complex deep learning model architecture, primarily designed for sequence processing in tasks such as sign language recognition. This model integrates multiple layers and operations to handle sequential input data effectively. The architecture begins with an InputLayer that accepts frames, formatted as a three-dimensional array [None, 128, 164], which then passes through a Masking layer to ignore certain types of input (e.g., padding or missing values) for the purpose of maintaining the integrity of the sequence processing. Subsequently, the data undergoes transformation in an Embedding layer, which expands the feature representation to 384 dimensions, aiding in richer feature extraction by the subsequent Encoder layer. This encoder processes the embedded input and outputs a condensed representation [None, 128, 256] to the Decoder, which aims to reconstruct or transform the sequence contextually, often used in language translation or similar tasks. Alongside, a separate InputLayer for phrases processes integer-encoded inputs, suggesting a possible multimodal approach combining textual and sequential input data. The decoder's output then feeds into a Sequential classifier, which finalizes the processing pipeline by generating predictions or classifications based on the learned features, outputting a processed signal [None, 32, 62]. This architecture indicates a sophisticated approach to handling complex patterns in sequence data, suitable for tasks requiring nuanced understanding of temporal dynamics and contextual dependencies.

IV. Experimental Results

Fig. 6 illustrates the key anatomical landmarks, or keypoints, used in the study for tracking and recognizing fingerspelling gestures in sign language. The diagram depicts a schematic of a human hand annotated with 21 distinct keypoints, each corresponding to critical joints and segments within the hand's structure. These keypoints include the wrist, the carpometacarpal (CMC), metacarpophalangeal (MCP),

proximal interphalangeal (PIP), distal interphalangeal (DIP) joints of each finger, and the tips of the fingers.

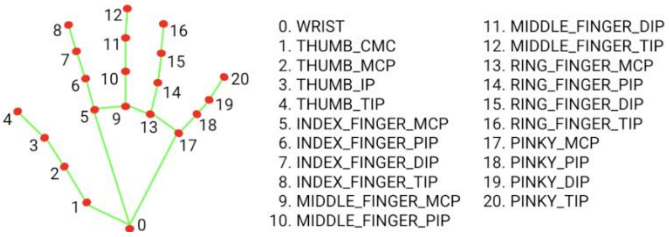

Fig. 6. Keypoints on the hand for fingerspelling.

The keypoints are numbered from 0 to 20, starting from the wrist and moving outward towards the fingertips. For instance, keypoint 0 represents the wrist, keypoint 1 the thumb CMC, keypoint 5 the index finger MCP, and so forth, culminating with keypoint 20 at the tip of the pinky finger. These annotations are crucial for machine learning models, which rely on these precisely defined points to accurately interpret and classify the gestures involved in fingerspelling.

The connections between the keypoints, represented by green lines, indicate the typical kinematic chains in hand anatomy essential for motion tracking and gesture recognition. These lines help in understanding how movements at one joint affect subsequent parts of the hand, which is vital for developing algorithms that can accurately interpret complex hand gestures. The clear labeling and structuring of these keypoints in the diagram provide a foundation for detailed analysis and discussion of the results related to fingerspelling recognition accuracy in the subsequent sections of the document.

Fig. 7 provides a detailed representation of the upper extremity keypoints utilized in the fingerspelling recognition algorithm, specifically highlighting the arrangement and connectivity of key anatomical landmarks across the fingers and wrist. This diagram focuses on the joints of the fingers and the wrist, essential for deciphering the precise configuration of hand gestures in sign language communication.

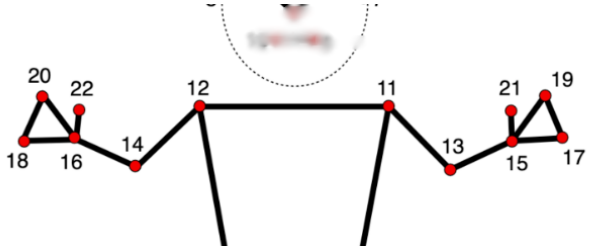

Fig. 7. Keypoints on the human body for fingerspelling.

In this schematic, keypoints are strategically positioned on the joints and tips of the fingers to capture the essential articulations necessary for sign language interpretation. The keypoints are numbered from 12 to 22, illustrating an array from the base of the wrist up through the tips of the fingers. Notably, keypoints 12 and 14 signify the wrist and the base of the fingers, respectively, forming a critical juncture from which the digital keypoints extend.

Each finger is represented by a sequence of keypoints, with the numbering extending outward towards the fingertips: the index finger from keypoint 14 to 16, the middle finger from 11 to 13, and so forth, with the additional articulations for more complex gestures indicated by keypoints 17, 19, 21, and 22. The lines connecting these points indicate the kinematic links that are essential for understanding how movements in one part of the hand influence the positioning and orientation of the other parts.

This configuration allows the machine learning models to track and interpret the dynamic and complex movements involved in sign language gestures, providing a robust framework for accurate gesture recognition. The clarity and layout of these keypoints are pivotal for analyzing the effectiveness of the fingerspelling recognition system, which is further explored in the results section of the study.

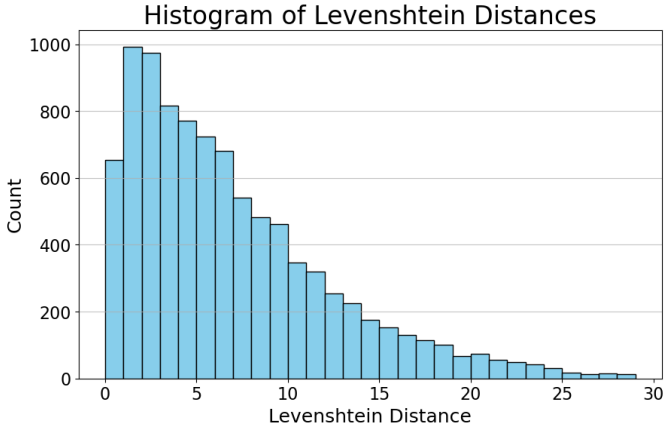

Fig. 8. Histogram of Levenshtain distances.

Fig. 8 presents a histogram of Levenshtein distances calculated from the output of the fingerspelling recognition system. The Levenshtein distance, a metric for measuring the difference between two sequences, is used here to evaluate the discrepancy between the predicted and actual spelled sequences in sign language communication. The x-axis of the histogram represents the Levenshtein distance, ranging from 0 to 30, while the y-axis displays the count of sequences falling into each distance category [47].

The distribution depicted in the histogram is skewed to the left, indicating that a majority of the sequence predictions by the model have a relatively low discrepancy from the target sequences, with the highest frequency observed in the range of 0 to 5. This suggests that the model is generally effective in accurately predicting the sequences, though errors increase as the distance values rise. The diminishing frequency as the distance increases confirms that fewer instances have larger errors, highlighting the effectiveness of the model in capturing the nuances of fingerspelling gestures with a considerable degree of accuracy. The shape and spread of the distribution provide crucial insights into the performance of the recognition system, revealing both its strengths in accurately recognizing many gestures and the areas where improvements might be necessary for those predictions exhibiting higher discrepancies.

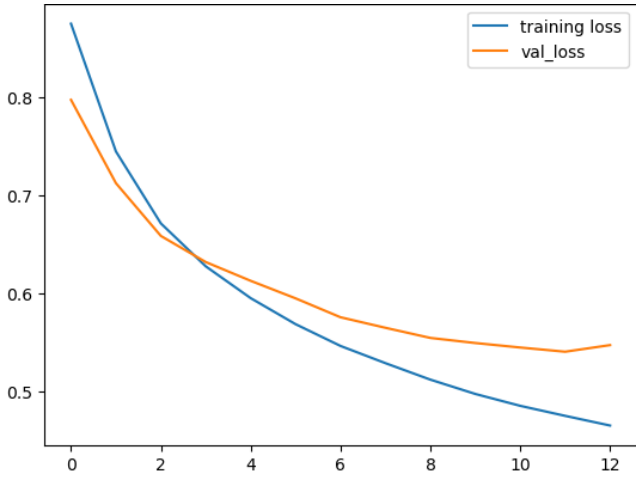

Fig. 9. Training and validation loss.

Fig. 9 displays a graphical representation of the training and validation loss curves for the sign language recognition model over a series of epochs. The x-axis of the graph indicates the number of epochs, while the y-axis represents the loss value, which quantifies the difference between the predicted outputs of the model and the actual target values during training and validation phases.

The blue line represents the training loss, illustrating how the model's error on the training set decreases as it learns from the data over successive epochs. The orange line, representing the validation loss, shows a similar decrease, indicating how well the model generalizes to new, unseen data. Both curves exhibit a steep decline in the initial epochs, signifying rapid learning and improvement in model performance.

As the number of epochs increases, both curves begin to plateau, suggesting that the model is approaching its optimal performance given the current architecture and data. The convergence of the training and validation loss lines toward the latter epochs indicates good model generalization without significant overfitting. This convergence is crucial for confirming that the model is not merely memorizing the training data but rather learning generalizable patterns that perform well on external data. The graph effectively underscores the learning dynamics of the model, highlighting areas where the training process is stable and effective, alongside pointing out the epochs after which learning saturates.

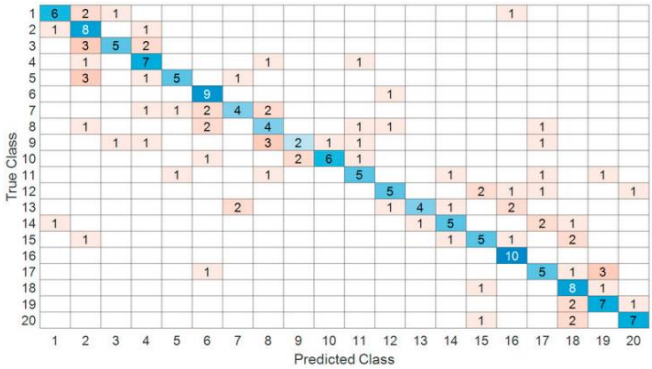

Fig. 10. Confusion matrix results.

Fig. 10 depicts a confusion matrix, a critical tool for evaluating the performance of the classification model developed for fingerspelling recognition. This matrix presents the counts of actual versus predicted class labels across a set of 20 classes, which represent different fingerspelling gestures.

The x-axis of the confusion matrix corresponds to the predicted class labels by the model, ranging from 1 to 20, while the y-axis represents the true class labels. Each cell within the matrix shows the number of instances that the model predicted a certain class (x-axis) for an actual class (y-axis). The diagonal cells, highlighted by darker shades, represent correct predictions where the predicted classes match the true classes. Off-diagonal cells indicate misclassifications, where the numbers denote how often a particular class was predicted instead of another.

A quick visual assessment of the matrix reveals several insights:

- The concentration of higher numbers along the diagonal line indicates good model accuracy for many classes, with prominent correctly classified instances such as in classes 1, 6, 7, and 9.
- Some classes, however, show notable confusion with others. For example, class 20 exhibits frequent misclassification, with incorrect predictions scattered across several other classes.
- Certain pairs of classes, such as 10 and 20 or 5 and 19, have higher confusion, suggesting similarities in the gestures that may be leading to these consistent misclassifications.

Overall, the confusion matrix provides a detailed view of the model's strengths and weaknesses across different fingerspelling gestures, highlighting specific areas where the model performs well and others where improvement is necessary. This visual tool is indispensable for diagnosing performance issues and guiding future enhancements to the model's classification capabilities.

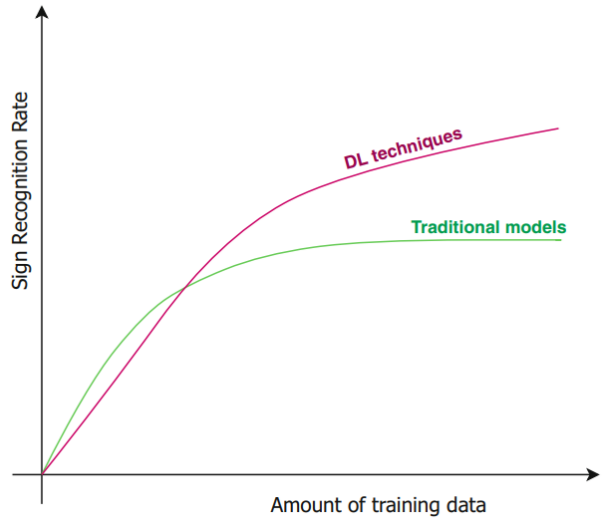

Fig. 11. Comparative performance of the proposed deep model vs. traditional models in sign language recognition as a function of training data volume.

Fig. 11 illustrates a comparative analysis of the performance between traditional machine learning models and deep learning (DL) techniques in the context of sign language recognition as a function of the amount of training data used. The graph plots the sign recognition rate on the vertical axis against the amount of training data on the horizontal axis. As depicted, both curves exhibit an increase in recognition rate with more data, demonstrating the typical behavior that more extensive training datasets generally improve the accuracy of predictive models. However, the curve representing deep learning techniques (colored in pink) is positioned above that of traditional models (colored in green), indicating a consistently higher recognition rate across the spectrum of data volumes. Notably, the deep learning curve shows a steeper initial ascent, suggesting that DL techniques are more effective at leveraging larger datasets to achieve significant improvements in accuracy [48]. This trend highlights the scalability and robustness of deep learning models in handling complex, high-dimensional data typical of sign language video inputs, compared to traditional models which plateau earlier and achieve lower peak recognition rates.

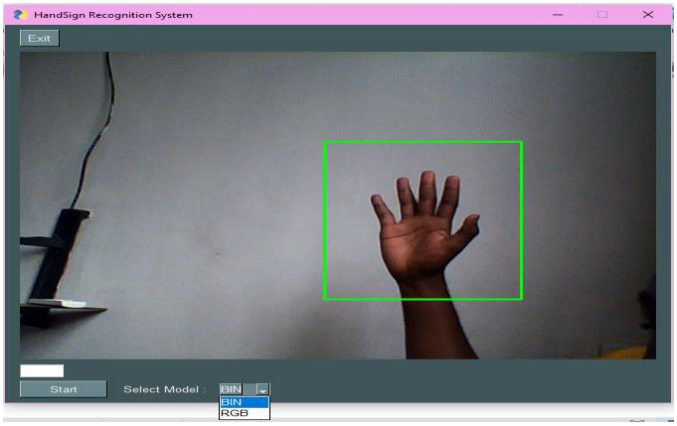

Fig. 12. Comparative performance of the proposed deep model vs. traditional models in sign language recognition as a function of training data volume.

Fig. 12 displays a screenshot of the HandSign Recognition System interface during operation, showcasing the system's ability to detect and highlight a hand gesture within a real-time video feed. The image illustrates a human hand positioned centrally against a plain background, with the hand's open gesture enclosed by a green bounding box, indicating successful detection by the system. This visual feedback is part of the user interface designed to allow users to verify the correct identification and tracking of hand gestures. The interface includes several operational controls such as “Start”, “Select Model”, along with options to switch between binary (“BIN”) and RGB color modes, enhancing the flexibility and usability of the system for different lighting conditions and background scenarios. The inclusion of such features underscores the system's practical application in real-world environments, where variability in input conditions can significantly affect performance.

V. Discussion

This research paper has examined the development and application of a deep learning-based sign language recognition system, which is instrumental in bridging communication gaps between the deaf and the hearing. The discussion elaborates on the implications of the findings, explores the challenges encountered, and suggests future research directions.

A. Efficacy of Deep Learning Models

The results demonstrated that deep learning techniques outperform traditional models in sign language recognition, particularly as the volume of training data increases. As depicted in Fig. 11, deep learning models exhibit a steeper improvement curve in recognition accuracy with the addition of training data. This can be attributed to the ability of deep learning models to extract complex features from high-dimensional data, a capability that traditional machine learning models lack. The advanced feature extraction allows for a more nuanced understanding of sign language gestures, which are inherently complex due to the variety of hand shapes, orientations, and movements involved [49].

B. System Performance in Real-Time Applications

Fig. 12 illustrates the practical application of the system in real-time environments. The system's capability to accurately detect and track hand gestures in real time underscores its potential for use in dynamic scenarios, such as live sign language translation or interactive educational tools. However, the performance in real-world conditions can be affected by factors such as lighting variations, background noise, and rapid movements. Although the current system handles these challenges to a certain extent, further refinement is necessary to improve robustness and ensure consistent performance regardless of external conditions.

C. Integration and Usability Challenges

The user interface, as shown in Fig. 12, is designed to be intuitive and user-friendly, offering essential controls like model selection and color mode adjustments. This is critical for ensuring that the system is accessible not only to researchers and professionals but also to lay users, including educators and individuals within the deaf community. However, the integration of such technology into everyday applications presents challenges, including the need for compatible hardware, user training, and ongoing support. Furthermore, there remains a significant need to develop standardized protocols for evaluating the usability of such systems in diverse settings [50].

D. Limitations and Scope for Improvement

While the system shows promising results, there are several limitations that need to be addressed. The current dataset, although extensive, may not fully represent the diversity within the global deaf community [51]. Sign language varies significantly not just internationally but also regionally; therefore, the system’s training on a more culturally and linguistically diverse dataset could enhance its applicability. Moreover, the confusion matrix in Fig. 10 reveals specific areas where the model confuses similar gestures. This could be mitigated by introducing more granular features and perhaps a temporal component to better differentiate between dynamically similar signs.

E. Future Directions

Looking forward, the research should focus on several key areas:

1) Data Diversification: Collecting and incorporating more diverse training data that cover a broader spectrum of sign languages and include more varied environments and lighting conditions.

2) Algorithm Optimization: Enhancing the model's architecture to improve its ability to learn from fewer data points, which is crucial for rare gestures or signs.

3) Real-Time processing improvements: Reducing latency further and increasing the processing speed to handle rapid sequences of gestures without delay.

4) User-Centric Design: Engaging with the deaf community to tailor the system's development to their needs and preferences, ensuring that the technology is both accessible and practical.

5) Cross-Platform compatibility: Ensuring the system is adaptable to various devices and platforms, enhancing its accessibility and practical utility.

The development of a sign language recognition system using deep learning techniques represents a significant technological advancement with the potential to impact real-world interactions profoundly. By continuously refining the system and addressing the outlined challenges, future iterations can provide even more reliable and inclusive communication tools for the deaf and hard-of-hearing communities.

VI. Conclusion

The research undertaken in this study has culminated in the development of an advanced sign language recognition system powered by deep learning techniques, showcasing significant potential to enhance communication between the deaf and hearing communities. The application of deep learning has been demonstrated to markedly outperform traditional models, especially as the volume of training data increases. This is evident in the system's enhanced ability to interpret complex hand gestures with high accuracy, addressing the dynamic and diverse nature of sign language. Our findings indicate that with sufficient training data, deep learning models can effectively capture the subtleties of sign language, which are often missed by more conventional approaches. The real-time operational capability of the system, as demonstrated, further underscores its practical utility in everyday applications, from educational settings to public services. However, challenges related to system integration, environmental variability, and data diversity call for ongoing improvements. Future research should aim to diversify the training datasets to include a broader array of sign languages and refine the system's robustness against external changes such as lighting and background variations. Engaging with the deaf community to tailor the technology to their needs will ensure that the advancements in sign language recognition technology are both practical and impactful. Ultimately, this research paves the way for creating more accessible and effective communication tools, fostering inclusivity and understanding across different sections of society.

References

[1] Shin, J., Miah, A. S. M., Akiba, Y., Hirooka, K., Hassan, N., & Hwang, Y. S. (2024). Korean Sign Language Alphabet Recognition through the Integration of Handcrafted and Deep Learning-Based Two-Stream Feature Extraction Approach. IEEE Access.

[2] Gao, Q., Ogenyi, U. E., Liu, J., Ju, Z., & Liu, H. (2020). A two-stream CNN framework for American sign language recognition based on multimodal data fusion. In Advances in Computational Intelligence Systems: Contributions Presented at the 19th UK Workshop on Computational Intelligence, September 4-6, 2019, Portsmouth, UK 19 (pp. 107-118). Springer International Publishing.

[3] Omarov, B., Batyrbekov, A., Suliman, A., Omarov, B., Sabdenbekov, Y., & Aknazarov, S. (2020, November). Electronic stethoscope for detecting heart abnormalities in athletes. In 2020 21st International Arab Conference on Information Technology (ACIT) (pp. 1-5). IEEE.

[4] Luqman, H. (2022). An efficient two-stream network for isolated sign language recognition using accumulative video motion. IEEE Access, 10, 93785-93798.

[5] Yin, L., Ying, H., & Meng-hao, Y. (2023). Chinese sign language recognition based on two-stream CNN and LSTM network. International Journal of Advanced Networking and Applications, 14(6), 5666-5671.

[6] Rastgoo, R., Kiani, K., & Escalera, S. (2022). Real-time isolated hand sign language recognition using deep networks and SVD. Journal of Ambient Intelligence and Humanized Computing, 13(1), 591-611.

[7] Omarov, B., Narynov, S., & Zhumanov, Z. (2023). Artificial Intelligence-Enabled Chatbots in Mental Health: A Systematic Review. Computers, Materials & Continua, 74(3).

[8] Kumar, E. K., Kishore, P. V. V., Kumar, M. T. K., & Kumar, D. A. (2020). 3D sign language recognition with joint distance and angular coded color topographical descriptor on a 2–stream CNN. Neurocomputing, 372, 40-54.

[9] Singla, N., Taneja, M., Goyal, N., & Jindal, R. (2023, March). Feature Fusion and Multi-Stream CNNs for ScaleAdaptive Multimodal Sign Language Recognition. In 2023 9th International Conference on Advanced Computing and Communication Systems (ICACCS) (Vol. 1, pp. 1266-1273). IEEE.

[10] Omarov, B., Altayeva, A., Demeuov, A., Tastanov, A., Kassymbekov, Z., & Koishybayev, A. (2020, December). Fuzzy controller for indoor air quality control: a sport complex case study. In International Conference on Advanced Informatics for Computing Research (pp. 53-61). Singapore: Springer Singapore.

[11] Al-Qurishi, M., Khalid, T., & Souissi, R. (2021). Deep learning for sign language recognition: Current techniques, benchmarks, and open issues. IEEE Access, 9, 126917-126951.

[12] Ghadami, A., Taheri, A., & Meghdari, A. (2024). A Transformer-Based Multi-Stream Approach for Isolated Iranian Sign Language Recognition. arXiv preprint arXiv:2407.09544.

[13] Subburaj, S., & Murugavalli, S. (2022). Survey on sign language recognition in context of vision-based and deep learning. Measurement: Sensors, 23, 100385.

[14] da Silva, D. R., de Araújo, T. M. U., do Rêgo, T. G., Brandão, M. A. C., & Gonçalves, L. M. G. (2024). A multiple stream architecture for the recognition of signs in Brazilian sign language in the context of health. Multimedia Tools and Applications, 83(7), 19767-19785.

[15] Miah, A. S. M., Hasan, M. A. M., Nishimura, S., & Shin, J. (2024). Sign language recognition using graph and general deep neural network based on large scale dataset. IEEE Access.

[16] Robert, E. J., & Duraisamy, H. J. (2023). A review on computational methods based automated sign language recognition system for hearing and speech impaired community. Concurrency and Computation: Practice and Experience, 35(9), e7653.

[17] Tao, T., Zhao, Y., Liu, T., & Zhu, J. (2024). Sign Language Recognition: A Comprehensive Review of Traditional and Deep Learning Approaches, Datasets, and Challenges. IEEE Access.

[18] Miah, A. S. M., Hasan, M. A. M., Jang, S. W., Lee, H. S., & Shin, J. (2023). Multi-stream graph-based deep neural networks for skeleton-based sign language recognition.

[19] Hamza, H. M., & Wali, A. (2023). Pakistan sign language recognition: leveraging deep learning models with limited dataset. Machine Vision and Applications, 34(5), 71.

[20] Patel, D. U., & Joshi, J. M. (2022). Deep leaning based static Indian-Gujarati Sign language gesture recognition. SN Computer Science, 3(5), 380.

[21] Bahia, N. K., & Rani, R. (2023). Multi-level taxonomy review for sign language recognition: Emphasis on indian sign language. ACM Transactions on Asian and Low-Resource Language Information Processing, 22(1), 1-39.

[22] Liu, T., Tao, T., Zhao, Y., Li, M., & Zhu, J. (2024). A signer-independent sign language recognition method for the single-frequency dataset. Neurocomputing, 582, 127479.

[23] Nihalani, R., Chouhan, S. S., Mittal, D., Vadula, J., Thakur, S., Chakraborty, S., ... & Saxena, A. (2024). Long Short-Term Memory (LSTM) model for Indian sign language recognition. Journal of Intelligent & Fuzzy Systems, (Preprint), 1-19.

[24] Chroni, E. T. (2024). Skeleton based approaches for isolated sign language recognition (Doctoral dissertation, Rutgers University-School of Graduate Studies).

[25] Bhaumik, G., Verma, M., Govil, M. C., & Vipparthi, S. K. (2022). ExtriDeNet: an intensive feature extrication deep network for hand gesture recognition. The Visual Computer, 38(11), 3853-3866.

[26] Doskarayev, B., Omarov, N., Omarov, B., Ismagulova, Z., Kozhamkulova, Z., Nurlybaeva, E., & Kasimova, G. (2023). Development of Computer Vision-enabled Augmented Reality Games to Increase Motivation for Sports. International Journal of Advanced Computer Science and Applications, 14(4).

[27] Xu, F., Chaudhary, L., Dong, L., Setlur, S., Govindaraju, V., & Nwogu, I. (2024, May). A Comparative Study of Video-Based Human Representations for American Sign Language Alphabet Generation. In 2024 IEEE 18th International Conference on Automatic Face and Gesture Recognition (FG) (pp. 1-6). IEEE.

[28] Sreemathy, R., Jagdale, J., Sayed, A. A., Ramteke, S. H., Naqvi, S. F., & Kangune, A. (2023, December). Recent works in Sign Language Recognition using deep learning approach-A Survey. In 2023 OITS International Conference on Information Technology (OCIT) (pp. 502-507). IEEE.

[29] Arslan, N. N., Şahin, E., & Akçay, M. (2023). Deep learning-based isolated sign language recognition: a novel approach to tackling communication barriers for individuals with hearing impairments. Journal of Scientific Reports-A, (055), 50-59.

[30] Hüseyinoğlu, A., Bilge, F. A., Bilge, Y. C., & Ikizler-Cinbis, N. (2024). Tinysign: sign language recognition in low resolution settings. Signal, Image and Video Processing, 1-10.

[31] Shin, J., Miah, A. S. M., Kabir, M. H., Rahim, M. A., & Shiam, A. A. (2024). A Methodological and Structural Review of Hand Gesture Recognition Across Diverse Data Modalities. arXiv preprint arXiv:2408.05436.

[32] Deng, Z., Leng, Y., Hu, J., Lin, Z., Li, X., & Gao, Q. (2024). SML: A Skeleton-based multi-feature learning method for sign language recognition. Knowledge-Based Systems, 112288.

[33] Núñez-Marcos, A., Perez-de-Viñaspre, O., & Labaka, G. (2023). A survey on Sign Language machine translation. Expert Systems with Applications, 213, 118993.

[34] Shah, S., Vaidya, J., Pipariya, K., & Shah, M. (2024). A Comprehensive Study on Relative Distances of Hand Landmarks Approach for American Sign Language Gesture. Augmented Human Research, 9(1), 1.

[35] Ilham, A. A., & Nurtanio, I. (2023). Dynamic Sign Language Recognition Using Mediapipe Library and Modified LSTM Method. International Journal on Advanced Science, Engineering & Information Technology, 13(6).

[36] Omarov, B., Suliman, A., Kushibar, K. Face recognition using artificial neural networks in parallel architecture. Journal of Theoretical and Applied Information Technology 91 (2), pp. 238-248. Open Access.

[37] GuruAkshya, C. (2024, April). Deep Learning Framework for Sign Language Recognition Using Inception V3 with Transfer Learning. In 2024 Third International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE) (pp. 1-6). IEEE.

[38] Hashi, A. O., Hashim, S. Z. M., & Asamah, A. B. (2024). A Systematic Review of Hand Gesture Recognition: An Update From 2018 to 2024. IEEE Access.

[39] Mohammadi, Z., Akhavanpour, A., Rastgoo, R., & Sabokrou, M. (2024). Diverse hand gesture recognition dataset. Multimedia Tools and Applications, 83(17), 50245-50267.

[40] Joy, T. S., Efat, A. H., Hasan, S. M., Jannat, N., Oishe, M., Mitu, M., & Fahim, A. M. (2023, December). Attention Trinity Net and DenseNet Fusion: Revolutionizing American Sign Language Recognition for Inclusive Communication. In 2023 26th International Conference on Computer and Information Technology (ICCIT) (pp. 1-6). IEEE.

[41] Zhou, Y., Xia, Z., Chen, Y., Neidle, C., & Metaxas, D. (2024, May). A multimodal spatio-temporal GCN model with enhancements for isolated sign recognition. In Proceedings of the {LREC-COLING} 2024 11th Workshop on the Representation and Processing of Sign Languages: Evaluation of Sign Language Resources. ELRA Language Resources Association (ELRA) and the International Committee on Computational Linguistics (ICCL).

[42] Wang, L., Ni, J., Gao, H., Li, J., Chang, K. C., Fan, X., ... & Yoo, C. (2023, July). Listen, Decipher and Sign: Toward Unsupervised Speech-to-Sign Language Recognition. In Findings of the Association for Computational Linguistics: ACL 2023 (pp. 6785-6800).

[43] Altayeva, A., Omarov, B., & Im Cho, Y. (2018, January). Towards smart city platform intelligence: PI decoupling math model for temperature and humidity control. In 2018 IEEE International Conference on Big Data and Smart Computing (BigComp) (pp. 693-696). IEEE.

[44] Liu, Y., Zhang, S., & Gowda, M. (2022). A practical system for 3-D hand pose tracking using EMG wearables with applications to prosthetics and user interfaces. IEEE Internet of Things Journal, 10(4), 3407-3427.

[45] Tursynova, A., Omarov, B., Sakhipov, A., & Tukenova, N. (2022). Brain Stroke Lesion Segmentation Using Computed Tomography Images based on Modified U-Net Model with ResNet Blocks. International Journal of Online & Biomedical Engineering, 18(13).

[46] Robinson, N., Tidd, B., Campbell, D., Kulić, D., & Corke, P. (2023). Robotic vision for human-robot interaction and collaboration: A survey and systematic review. ACM Transactions on Human-Robot Interaction, 12(1), 1-66.

[47] Tursynova, A., Omarov, B., Tukenova, N., Salgozha, I., Khaaval, O., Ramazanov, R., & Ospanov, B. (2023). Deep learning-enabled brain stroke classification on computed tomography images. Comput. Mater. Contin, 75(1), 1431-1446.

[48] Fragkiadakis, M. (2024). LOT, msterdam. from https://hdl. handle. net/1887/3734159 Version: Publisher's Version License: Licence agreement concerning inclusion of doctoral thesis in the Institutional Repositor of the Uni ersit of Leiden Downloaded from: https://hdl. handle. net/1887/3734159.

[49] Onalbek, Z. K., Omarov, B. S., Berkimbayev, K. M., Mukhamedzhanov, B. K., Usenbek, R. R., Kendzhaeva, B. B., & Mukhamedzhanova, M. Z. (2013). Forming of professional competence of future tyeacher-trainers as a factor of increasing the quality. Middle East Journal of Scientific Research, 15(9), 1272-1276.

[50] Omarov, B., Orazbaev, E., Baimukhanbetov, B., Abusseitov, B., Khudiyarov, G., & Anarbayev, A. (2017). Test battery for comprehensive control in the training system of highly Skilled Wrestlers of Kazakhstan on national wrestling" Kazaksha Kuresi". Man In India, 97(11), 453-462.

[51] Jiang, X., Zhang, Y., Lei, J., & Zhang, Y. (2024). A Survey on Chinese Sign Language Recognition: From Traditional Methods to Artificial Intelligence. CMES-Computer Modeling in Engineering & Sciences, 140(1).