HANOI UNIVERSITY OF SCIENCE AND TECHNOLOGY
THE SCHOOL OF INFORMATION AND COMMUNICATION TECHNOLOGY
ICT K67

IT3160E
INTRODUCTION TO ARTIFICIAL INTELLIGENCE

SIGN LANGUAGE DETECTION

PROJECT REPORT

MEMBER LIST

Nguyễn Đình Dương	20225966	duong.nd225966@sis.hust.edu.vn
Nguyễn Mỹ Duyên	20225967	duyen.nm225967@sis.hust.edu.vn
Hồ Bảo Thư	20226003	duong.nd225966@sis.hust.edu.vn
Hà Việt Khánh	20225979	khanh.hv225979@sis.hust.edu.vn
Nguyễn Trọng Minh Phương	20225992	phuong.ntm5992@sis.hust.edu.vn

Teacher: Phạm Văn Hải

Hà Nội — 2024

CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Our team research an innovative gestural communication tool designed to address communication barriers, particularly for individuals with hearing impairments. The tool translates sign language into text or speech in real-time, offering efficient and accurate communication solutions. Incorporating advanced AI models like the Random Forest Classifier and MediaPipe Hands, the system ensures precise hand gesture detection and sign language interpretation. With anticipated accuracy rates of 90% to 95%, the tool sets a new standard for precision in sign language translation. Emphasizing intuitive design principles, the tool prioritizes usability and accessibility, fostering inclusivity and empowerment in communication. The initiative aims to create a more inclusive world by providing individuals with hearing impairments equal access to effective communication tools and opportunities for meaningful engagement in society.

1 PROBLEM DETECTION

According to the World Health Organization (WHO), more than 5% of the global population, totaling approximately 466 million individuals, suffer from disabling hearing loss. Regrettably, only a minority among them possess proficiency in sign language—a visual-based natural language reliant on hand gestures, movements, facial expressions, and body language. Consequently, this situation gives rise to three primary challenges:

1. **Communication Barriers:** Individuals with hearing loss who lack proficiency in sign language often face significant challenges in communicating with others. This can lead to feelings of isolation, exclusion, and frustration in personal and professional contexts. Without effective communication channels, accessing information, expressing oneself, and engaging in social interactions become formidable tasks.
2. **Educational Disparities:** The lack of access to sign language or other communication accommodations in educational settings can result in educational disparities for individuals with hearing loss. This may manifest in difficulties understanding classroom lectures, participating in group discussions, accessing instructional materials, and receiving adequate support from educators. As a consequence, academic performance and educational outcomes may be adversely affected, potentially limiting opportunities for future advancement.
3. **Employment Discrimination:** In the workplace, individuals with hearing loss who do not have proficiency in sign language may encounter discriminatory practices and barriers to employment. Employers may overlook their qualifications or hesitate to hire them due to concerns about communication barriers or the perceived need for accommodations. As a result, individuals with hearing loss may face limited job opportunities, reduced career advancement prospects, and unequal treatment in the workplace, contributing to socioeconomic inequalities.

Addressing these challenges requires concerted efforts to promote accessibility, inclusion, and equal opportunities for individuals with hearing loss. This involves implementing policies and practices that support the use of sign language, providing appropriate accommodations in educational and workplace settings, raising awareness about the needs of individuals with hearing loss, and fostering a culture of inclusivity and acceptance.

2 PURPOSE

The purpose of this initiative is multifaceted, aiming to address the communication challenges faced by individuals with hearing impairments. Through tailored assistance and support mechanisms, our primary objective is to enhance the communicative abilities of those with hearing impairments, ensuring their messages are comprehensively understood by others. By bridging the gap between the communication methods of deaf individuals, whether through speech or writing, and the understanding of their intended audience, we aspire to facilitate seamless interaction in various contexts.

Moreover, this initiative seeks to empower individuals with hearing impairments, enabling them to articulate their thoughts, emotions, and ideas with clarity and precision. By providing the tools and resources necessary for effective expression, we endeavor to amplify their voices within their familial, educational, and societal spheres. Through advocacy and education, we strive to cultivate an inclusive environment where the unique perspectives and experiences of deaf and hearing-impaired individuals are acknowledged, respected, and integrated into broader social discourse.

In essence, the purpose of this initiative extends beyond mere facilitation of communication; it embodies a commitment to promoting equity, dignity, and empowerment for individuals with hearing impairments. By fostering understanding, acceptance, and appreciation of diverse modes of communication, we endeavor to create a more inclusive and accessible society for all.

3 INPUTS AND OUTPUTS

3.1 Inputs

In this section, we outline the inputs and outputs of our system. Our inputs primarily consist of video clips and live video feeds containing hand gestures, captured through various devices such as cameras or smartphones. These inputs serve as the raw data that our system processes to extract meaningful information.

Our dataset comprises images of hand signs, publicly available on Kaggle. This dataset includes 26 different characters, with each character accompanied by approximately 3000 images. These characters represent the alphabet from A to Z in sign language. This diverse dataset enables us to train our models effectively, capturing the intricacies and variations present in real-world hand gestures.
Href: asl-alphabet data and Sign-language-gesture-images-dataset

3.2 Outputs

The main output of our system is the conversion of these hand gestures into either text or speech. Through sophisticated algorithms and machine learning techniques, we analyze the input videos to recognize and interpret the hand gestures performed. Subsequently, we generate corresponding textual representations or convert them into spoken language, depending on the application's requirements.

By transforming hand gestures into text or speech, our system enables individuals with

Figure 1: Input Data

Figure 2: Hand sign language

hearing impairments to access and comprehend information conveyed through visual communication modalities. This facilitates greater inclusivity and accessibility in communication, empowering individuals to participate more fully in various social, educational, and professional settings.

4 State of Art

Researchers and developers have been actively engaged in creating and refining automatic sign language recognition systems, leveraging cutting-edge techniques in machine learning, computer vision, and natural language processing. These systems hold immense potential for facilitating real-time translation of sign language into spoken or written language, enhancing accessibility in various domains including education, communication, and entertainment. Furthermore, efforts are underway to ensure the accuracy, robustness, and inclusivity of these systems across diverse signing styles and environments. As such, the landscape of sign language recognition technology continues to evolve rapidly, driven by a commitment to accessibility and inclusivity for all. There are many sign language recognition technologies that have made breakthroughs recently.

4.1 Natural Language-Assisted Sign Language Recognition (1)

Sign languages are visual languages that convey information through the movements of the hands, body, head, mouth, and eyes, making them completely separate and distinct from natural languages.Due to the inherent constraints imposed by the combinations of visual elements, there is a considerable number of visually indistinguishable signs (VISigns) in sign languages. This characteristic poses limitations on the recognition capabilities of visual neural networks. To address this challenge, we introduce the Natural Language-Assisted Sign Language Recognition (NLA-SLR) framework. This framework leverages the semantic

information embedded within glosses (sign labels).

Firstly, when VISigns exhibit similar semantic meanings, we employ a language-aware label smoothing approach. This method involves replacing hard labels with soft ones, akin to the well-established technique. For each training video, we adopt fastText to generate smoothed labels, thereby enhancing the model's ability to discern subtle semantic differences.

Conversely, in cases where VISigns convey distinct semantic meanings, we utilize the inter-modality mixup technique. This innovative approach involves independently integrating the vision feature with each gloss feature to create a blended representation. By seamlessly merging visual and semantic cues, this method enriches the model's understanding of the diverse semantic nuances inherent in sign language gestures.

The datasets had been used including MSASL, WLASL, and NMFs-CSL.In this study, S3D is opted as the backbone of VKNet. This decision is rooted in S3D's exceptional balance between accuracy and processing speed.

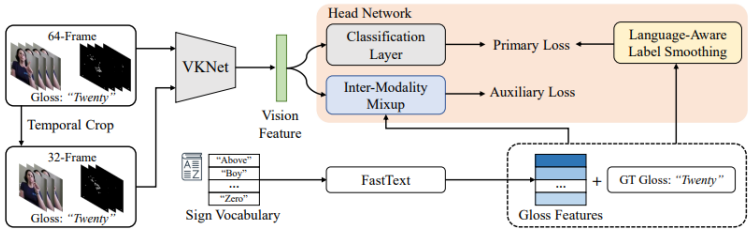

Figure 3: An overview of NLA-SLR 3 (1)

In data pre-processing, we Use HRNet trained on COCO-WholeBody, 63 key points (11 for upper body, 10 for mouth, and 42 for two hands)/frame. Besides, we also introduce a novel backbone, video-keypoint network, which not only models both RGB videos and human body keypoints but also derives knowledge from sign videos of different temporal re-ceptive field.

The network architectures of VKNet-32 and VKNet-64 are identical. Each of them employs a two-stream architecture comprising a video encoder and a keypoint encoder. Specifically, given a training sample belonging to the b-th class, label smoothing replaces the one-hot label with a soft label $y \in \mathbb{R}^N$.The head network follows the algorithm:

$$y[i] = \begin{cases} 1-\epsilon & \text{if } i = b, \\ \epsilon . \frac{\exp\left(\frac{s[i]}{\tau}\right)}{\sum_{i=1, i\neq b}^{N} \exp\left(\frac{s[i]}{\tau}\right)} & \text{otherwise.} \end{cases}$$

where τ denotes a temperature parameter. The classification loss L_{CLS} is a simple cross-entropy loss applied on the prediction and soft label y.

The accuracy of this algorithm, using both Lang-LS and sign mixup along with the VKNet can achieve the best performance: 61.05%/91.45% on the top-1 and top-5 accuracy, respectively.

4.2 Using reptile search algorithm with hybrid deep learning (2)

One of challenges with SLR comprises the variability in sign language through various cultures and individuals, the difficulty of certain signs, and require for realtime processing. This study introduces an Automated Sign Language Detection and Classification using Reptile Search Algorithm with Hybrid Deep Learning (SLDC-RSAHDL).

The presented SLDC-RSAHDL technique detects and classifies different types of signs using DL and metaheuristic optimizers. In the SLDC-RSAHDL technique, MobileNet feature extractor is utilized to produce feature vectors, and its hyperparameters can be adjusted by manta ray foraging optimization (MRFO) technique. The mathematical model written as:

$$\widehat{G} = \sum_{m} \sum_{i,j} \widehat{K}_{i,j,m} \, F_{k+1-1,l+j-1,m}$$

$$G_{k,l,n} = \sum_{m} \widehat{G}_{k,l,m} \cdot \overline{K}_{m,n}$$

The foraging chain has been developed if manta rays arrange head-to-tail. In each iteration, an optimum solution was utilized for updating every individual. The subsequent mathematical model can demonstrate it:

$$x_i^d(t+1) = \begin{cases} x_i^d(t) + r.(x_{best}^d(t) - x_{best}^d(t)) + \alpha.(x_{best}^d(t) - x_i^d(t)) & , i = 1 \\ x_i^d(t) + r.(x_{i-1}^d(t) - x_i^d(t)) + \alpha.(x_{best}^d(t) - x_i^d(t)) & i = 2, 3, ..., N. \end{cases}$$

For sign language classification, the SLDC-RSAHDL technique applies HDL model, which incorporates the design of Convolutional Neural Network (CNN) and Long-Short Term Memory (LSTM). At last, the RSA was exploited for the optimal hyperparameter selection of the HDL model, which resulted in an improved detection rate.

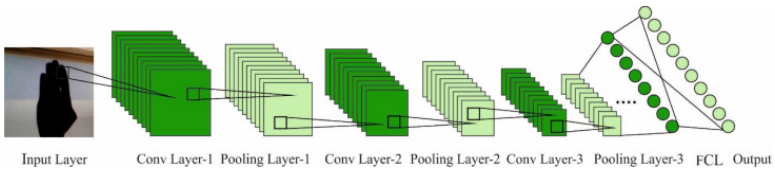

Figure 4: Sign language classification using optimal HDL model(2)

The RSA method creates a fitness function (FF) to make superior classifier result. It explains a positive integer to exemplify the good performance of candidate outcomes. During this effort, the minimizing of classifier error rate was supposed that FF is formulated in:

$$fitness(x_i) = ClassifierErrorRate(x_i) = \frac{number\ of\ misclassified\ samples}{Total\ number\ of\ samples} * 100$$

Based on the evaluation metrics, it can be observed that the algorithm achieves a very high performance in classification. With an accuracy rate of 99.51%, along with precision and recall scores of 99.42% and 99.43% respectively, we can conclude that the algorithm performs well in classifying the samples. This is further illustrated by the high F-score value of 99.43%, indicating that the algorithm provides a good balance between precision and recall.

5 AI Models and Algorithms

5.1 Theoretical basis

5.1.1 Random Forest Classifier

Random forests are an Ensemble method, meaning they combine predictions from other models called decision trees. During the training phase, multiple trees are created using different random subsets of the data and features. This randomness introduces variability among individual trees, reducing the risk of overfitting and improving overall prediction performance. Each decision tree is like an expert, providing its opinion on how to classify the data. In prediction, the algorithm aggregates the results of all trees, either by voting (for classification tasks) or by averaging (for regression tasks).

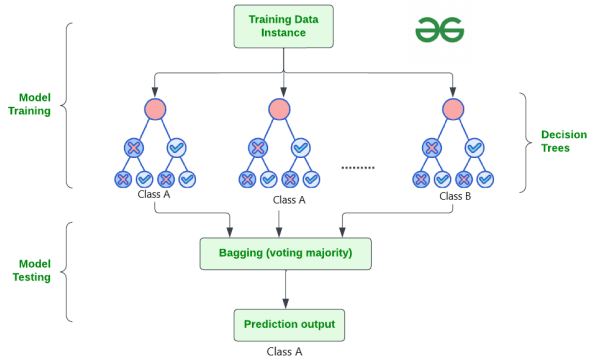

Figure 5: Random forest classifier (3)

The technique of bagging is a cornerstone of Random Forest's training strategy which involves creating multiple bootstrap samples from the original dataset, allowing instances to be sampled with replacement. This results in different subsets of data for each decision tree, introducing variability in the training process and making the model more robust.
Random forests find utility in a variety of industries, from predicting disease in healthcare to analyzing consumer behavior in business, thanks to their ability to manage complexity, reducing excessive input on, and because it provides reliable results.

5.1.2 Convolutional Neural Network (CNN)

A Convolutional Neural Network (CNN) is a type of artificial neural network specially designed for processing structured grid data, such as images or videos. It's particularly powerful in tasks involving visual imagery, like image recognition, classification, segmentation, and even generation.

Here's a breakdown of its components:

1. **Convolutional Layers**: These layers apply convolutional filters (also known as kernels) to the input data. These filters detect features in the input, such as edges, textures, or patterns, by sliding over the input data and performing element-wise multiplication and summation.

2. **Pooling Layers**: After convolutional layers, pooling layers are often applied to reduce the spatial dimensions of the feature maps produced by the convolutional layers. This helps in reducing the computational complexity and controlling overfitting.
3. **Activation Functions**: Non-linear activation functions, such as ReLU (Rectified Linear Unit), are applied after convolutional and pooling layers to introduce non-linearity into the network and enable it to learn complex patterns.
4. **Fully Connected Layers**: Following several convolutional and pooling layers, fully connected layers are often used to perform classification based on the features extracted by the earlier layers. These layers connect every neuron in one layer to every neuron in the next layer.
5. **Output Layer**: The output layer produces the final output of the network, which could be class probabilities in classification tasks or pixel values in image generation tasks.

CNNs are trained using labeled data through a process called backpropagation, where the network learns to adjust its parameters (weights and biases) to minimize the difference between its predictions and the actual labels. Training typically involves feeding forward input data through the network, comparing the output to the ground truth labels using a loss function, and then using an optimization algorithm like gradient descent to update the network parameters iteratively.

Overall, CNNs have revolutionized the field of computer vision and are widely used in various applications, including object detection, facial recognition, medical image analysis, and autonomous vehicles, among others.

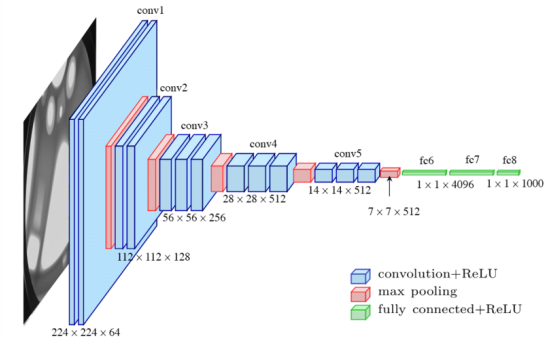

Figure 6: CNN's layers (4)

5.2 Models

In this section, we present two distinct AI models/algorithms, denoted as Type 1 and Type 2, each serving specific functions within our system.

5.2.1 Random Forest Classifier

Type 1 utilizes a combination of the Random Forest Classifier and MediaPipe Hands AI model, an AI model integrated into the MediaPipe library. (5)
MediaPipe Hands specializes in tracking within video frames with the ability to perceive the shape and motion of hands. It employs machine learning (ML) to infer the keypoint localization of 21 hand-knuckle coordinates within the detected hand regions as below. The model was trained on approximately 30K real-world images, as well as several rendered synthetic hand models imposed over various backgrounds. Since running the palm detection model

is time consuming, when in video or live stream running mode, Hand Landmarker uses the bounding box defined by the hand landmarks model in one frame to localize the region of hands for subsequent frames. Hand Landmarker only re-triggers the palm detection model if the hand landmarks model no longer identifies the presence of hands or fails to track the hands within the frame. This reduces the number of times Hand Landmarker triggers the palm detection model.

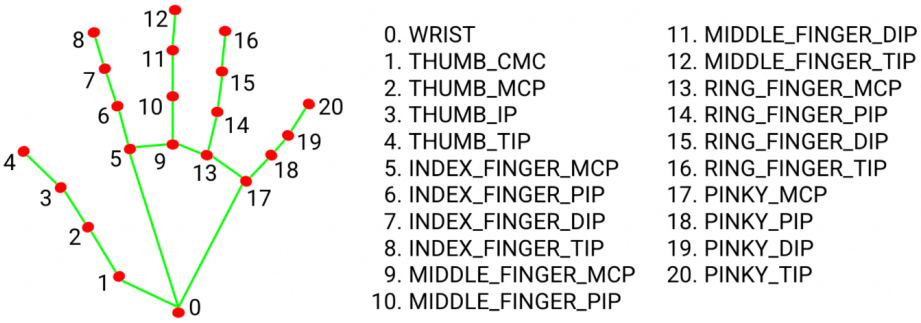

Figure 7: 21 hand landmarks (5)

Finally, the data is extracted and processed by the Random Forest Classifier to generate predictions.

The results clearly indicate that Type 1 excels in processing visual data by accurately detecting hand landmarks. This precision offers a distinct advantage, particularly in tasks requiring detailed analysis of hand motions.

5.2.2 VGG16 Convolutional Neural Network (CNN)

Type 2 leverages the capabilities of the VGG16 Convolutional Neural Network (CNN), developed by the research team at the University of Oxford (7).It is characterized by its depth, consisting of 16 layers, including 13 convolutional layers and 3 fully connected layers. These layers are organized into blocks, with each block containing multiple convolutional layers followed by a max-pooling layer for downsampling.This architecture include the use of ReLU activation function and the final fully connected layer outputting probabilities for 1000 classes using softmax activation.

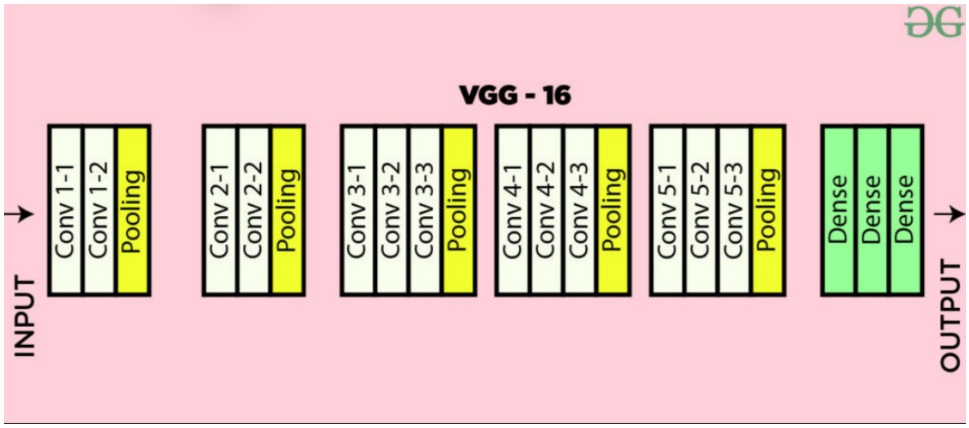

Figure 8: VGG-16 architecture Map (6)

VGG-16 is renowned for its simplicity and effectiveness, as well as its ability to achieve strong performance on various computer vision tasks, including image classification and object recognition. Through its convolutional layers, VGG16 learns hierarchical features, progressing from low-level features like edges and corners to high-level features such as

complex shapes and objects. This hierarchical feature extraction enables VGG16 to comprehend the spatial structure of images comprehensively, facilitating accurate classification and analysis within our system.

6 Expected Results

Our anticipated outcomes are structured to achieve significant advancements in gestural communication technology:

1. Integration of both image and real-time video data streams to enrich gestural communication capabilities.
2. Substantial enhancement in the speed and efficiency of sign language translation processes.
3. Attainment of a high accuracy level ranging between 80% to 95% in the recognition and interpretation of hand gestures.
4. Development of a user-friendly interface, meticulously crafted with adherence to fundamental design principles, ensuring seamless usability and accessibility for all users.

These expected results underscore our commitment to delivering a sophisticated and effective gestural communication solution that addresses the diverse needs of our users while maintaining high standards of performance and usability.

6.1 Experiment

Our experiment focuses on rigorous testing and validation procedures to ensure the robustness and reliability of our gestural communication system. The experiment encompasses two key aspects:

6.1.1 Data Pipeline Integrity

The first component of our experiment is dedicated to verifying the integrity and functionality of the data pipeline. Through systematic checks and validation procedures, we meticulously examine each stage of the data pipeline to ensure that data integrity is maintained and that there are no instances of corruption. Upon detecting any anomalies or inconsistencies, corrective measures are promptly implemented to rectify the issues and restore the data pipeline to its intended state. The corrected data is then validated to ensure that it aligns with the expected format and quality standards.

6.1.2 Model Testing

The second component of our experiment involves comprehensive testing of our AI models to assess their performance and accuracy. This testing encompasses various aspects, including:

- **Invariance Test:** Evaluating the model's ability to maintain consistent performance across different conditions and environments.

- **Directional Expectation Test:** Assessing the model's capability to accurately interpret and respond to directional gestures.
- **Bias/Fairness Test:** Analyzing the model's behavior to identify and mitigate any biases or unfairness in its decision-making process.
- **Model Output Consistency:** Ensuring that the model consistently produces reliable and coherent output across different inputs and scenarios.

By conducting thorough testing and validation procedures, we aim to ascertain the effectiveness and reliability of our gestural communication system, thereby instilling confidence in its performance and capabilities.

6.2 Evaluation Metrics

In evaluating the performance of our gestural communication system, we employ a set of comprehensive metrics to assess its accuracy and effectiveness. These metrics provide valuable insights into the system's performance across various dimensions:

6.2.1 Accuracy

Accuracy measures the overall correctness of our system by calculating the proportion of correctly classified instances among all instances. It serves as a fundamental indicator of the system's ability to make accurate predictions and classifications.

6.2.2 Precision

Precision assesses the quality of positive predictions made by our system. It calculates the proportion of correctly predicted positive instances among all instances predicted as positive. Precision is crucial for evaluating the reliability and trustworthiness of positive predictions.

6.2.3 Recall

Recall measures the system's ability to capture all positive instances within the dataset. It calculates the proportion of correctly predicted positive instances among all actual positive instances. Recall is essential for evaluating the system's completeness and ability to identify all relevant instances.

6.2.4 F1-Score

The F1-score provides a balanced assessment of the system's performance by considering both precision and recall. It calculates the harmonic mean of precision and recall, offering a single metric to evaluate the model's performance comprehensively. The F1-score enables us to assess the trade-off between precision and recall, providing valuable insights into the overall effectiveness of our gestural communication system.

By employing these evaluation metrics, we aim to conduct a thorough and rigorous assessment of our system's performance, ensuring its reliability and effectiveness in real-world applications.

7 Results

7.1 Random Forest Classifier

7.1.1 System Infrastructure

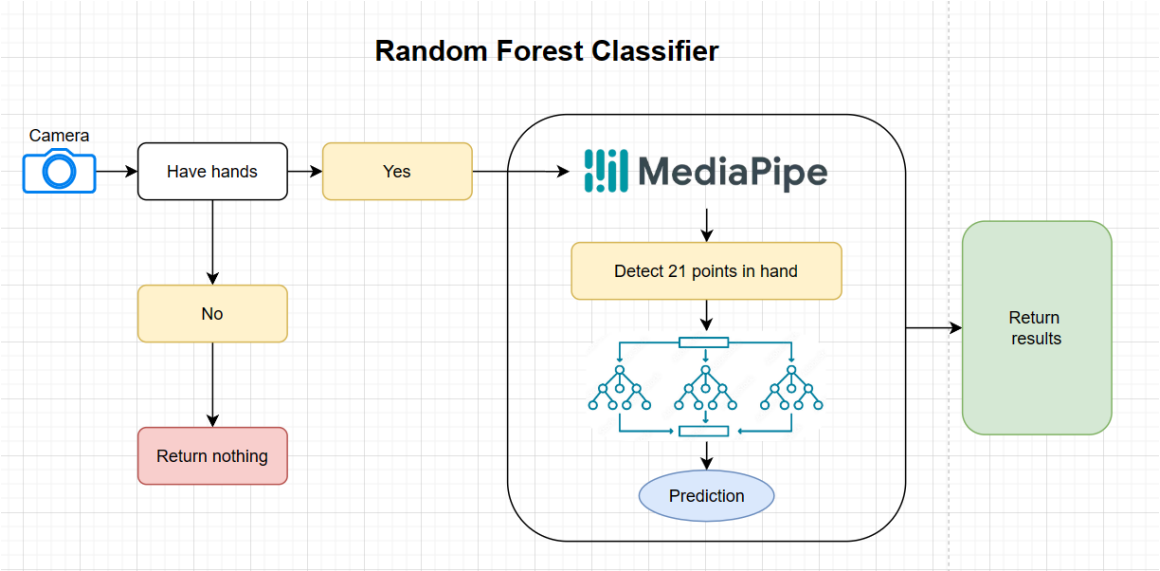

Figure 9: System Infrastructure of Type 1

7.1.2 Demo/interface

The following is a demo image that we have deployed, utilizing the rear camera. Afterwards, the system will automatically detect the hand and provide predictions.

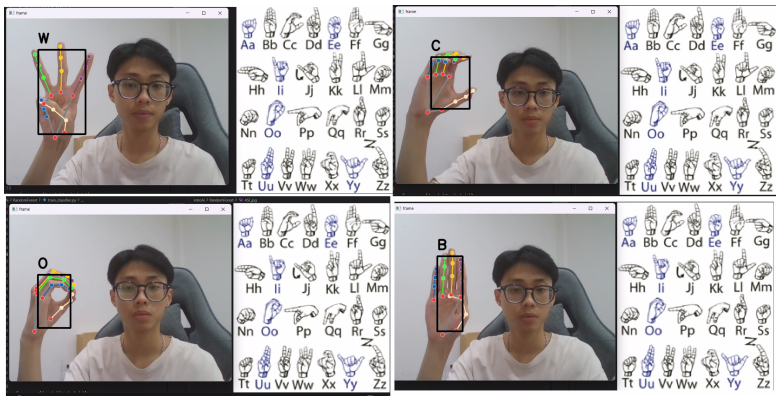

Figure 10: Interface of Random Forest Classifier Model

7.1.3 Evalution

Evalution of First Dataset
After trainning models, we evaluate the models with some aspect:

Class	Precision	Recall	F1-Score	Support
A	1.00	1.00	1.00	139
B	0.99	1.00	1.00	145
C	0.99	1.00	0.99	148
D	1.00	1.00	1.00	180
E	1.00	0.99	1.00	154
F	1.00	0.99	1.00	180
G	0.99	1.00	1.00	174
H	0.99	0.99	0.99	180
I	1.00	1.00	1.00	180
J	0.98	1.00	0.99	180
K	1.00	1.00	1.00	180
L	1.00	1.00	1.00	180
M	1.00	0.98	0.99	154
N	0.99	1.00	1.00	103
O	1.00	1.00	1.00	180
P	1.00	0.96	0.98	157
Q	0.96	1.00	0.98	180
R	0.99	0.98	0.99	180
S	0.99	0.99	0.99	180
T	1.00	1.00	1.00	180
U	0.98	0.95	0.96	172
V	0.97	1.00	0.99	169
W	0.99	1.00	1.00	179
X	1.00	0.99	0.99	169
Y	0.99	1.00	1.00	180
Z	0.99	0.99	0.99	180
Accuracy			0.99	4383
Macro Avg	0.99	0.99	0.99	4383
Weighted Avg	0.99	0.99	0.99	4383

Table 1: Classification Report

Accuracy: 0.9932
Support: 4383
From the output above, several observations can be inferred:

1. **Overall Accuracy** The model achieves an overall accuracy of approximately 99.41%, indicating that the model performs very well on the test dataset.
2. **Classification Report:**
    - Most classes have high accuracy, with precision, recall, and f1-score close to 1, suggesting that the model effectively classifies these classes.
    - However, some classes have lower scores such as classes M, N, P, U, V, indicating a need for further investigation into potential issues.

Evalution of Second Dataset

Table 2: Classification Report

Class	Precision	Recall	F1-Score	Support
0	1.00	1.00	1.00	31
1	1.00	1.00	1.00	21
2	0.95	1.00	0.97	53
3	1.00	1.00	1.00	27
4	0.98	1.00	0.99	100
5	1.00	1.00	1.00	100
6	1.00	0.94	0.97	31
7	0.98	1.00	0.99	44
8	1.00	1.00	1.00	81
9	1.00	1.00	1.00	36
A	1.00	0.99	0.99	100
B	1.00	1.00	1.00	100
C	1.00	1.00	1.00	100
D	1.00	1.00	1.00	100
E	1.00	1.00	1.00	16
F	1.00	1.00	1.00	100
G	1.00	1.00	1.00	100
H	1.00	1.00	1.00	100
I	1.00	1.00	1.00	100
J	1.00	1.00	1.00	100
K	1.00	1.00	1.00	100
L	1.00	1.00	1.00	100
M	1.00	1.00	1.00	3
N	1.00	1.00	1.00	100
O	1.00	1.00	1.00	14
P	1.00	1.00	1.00	100
Q	1.00	1.00	1.00	100
R	1.00	1.00	1.00	100
S	0.90	1.00	0.95	18
T	1.00	0.95	0.98	87
U	1.00	1.00	1.00	100
V	1.00	0.99	0.99	100
W	1.00	1.00	1.00	100
X	1.00	1.00	1.00	100
Y	1.00	1.00	1.00	50
Z	1.00	1.00	1.00	100
_	1.00	1.00	1.00	100
Accuracy			1.00	2812
Macro Avg	0.99	1.00	1.00	2812
Weighted Avg	1.00	1.00	1.00	2812

99.53769559032717% of samples were classifier correctly !
Accuracy: 0.9953769559032717
Overall Performance

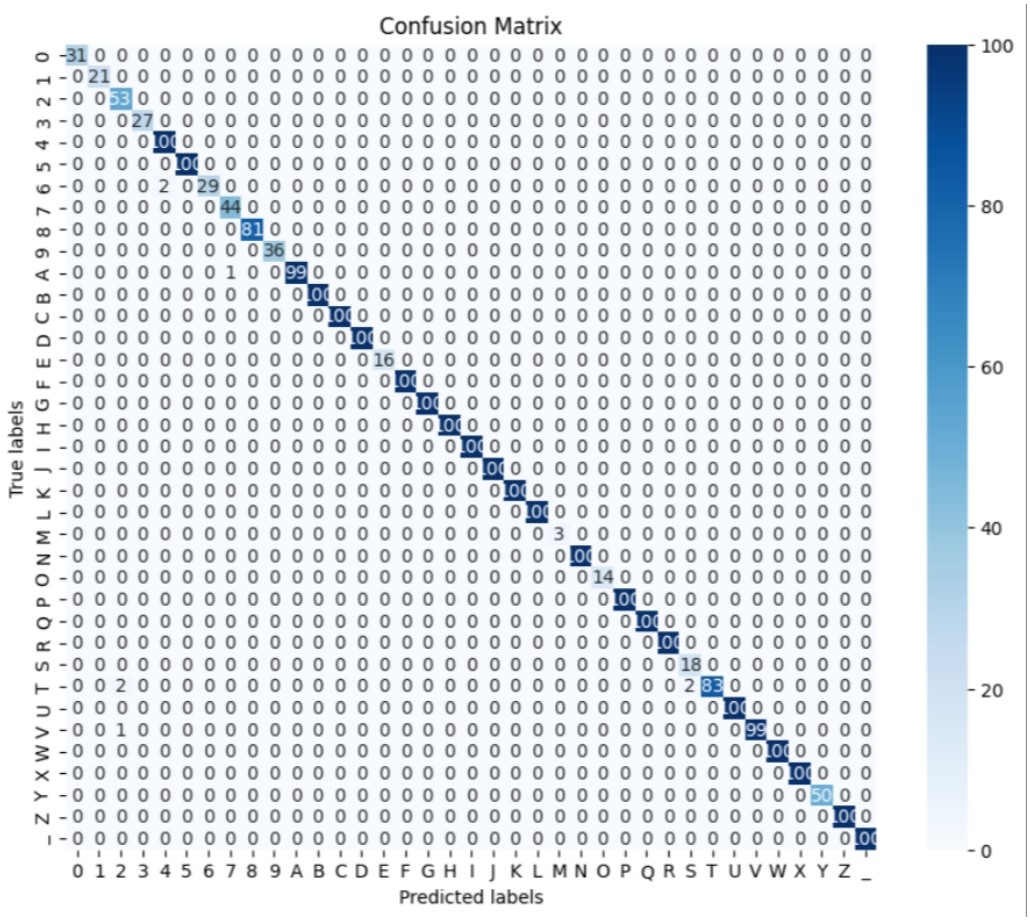

Figure 11: Confussion Matrix of Type 1

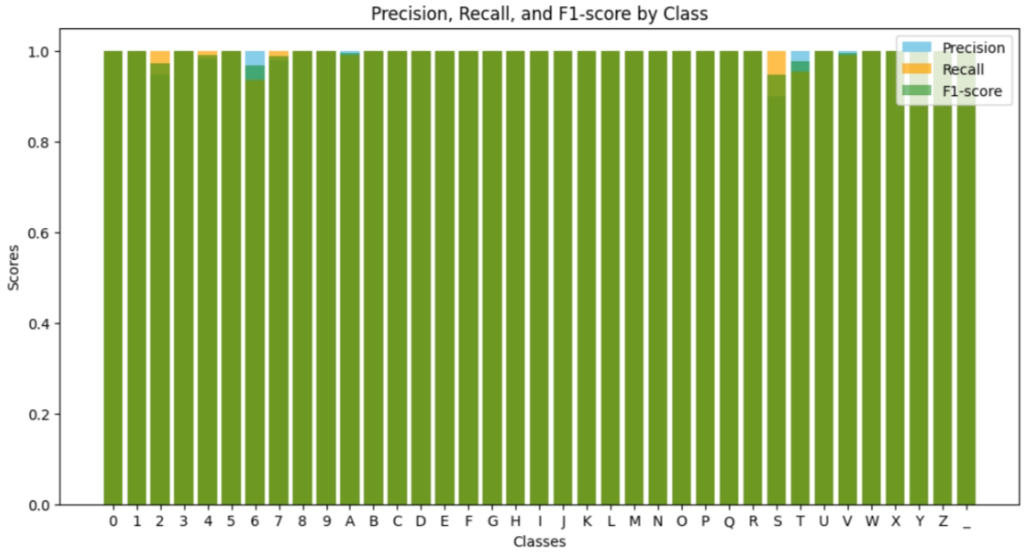

Figure 12: Precision, Recall, and F1-score of Type 1

- **Accuracy**: The overall accuracy of the model is approximately 99.72%, indicating that the model correctly classified nearly all samples in the test set. This high level of accuracy suggests that the model is very effective at distinguishing between the different classes.
- **Precision, Recall, and F1-Score**:
  - Most classes have perfect scores of 1.00 for precision, recall, and F1-score, indicating that the model performs exceptionally well across these classes.

  - Classes such as '2', '4', '6', '7', 'T', and 'V' have slightly lower but still very high scores, demonstrating minor variability in performance but still indicating strong classification ability.

- **Support**:
  - The support values indicate the number of true instances for each label in the test set. While most classes have 100 instances, some classes like '0', '1', '2', '3', '6', '7', '8', '9', 'E', 'M', 'O', 'S', 'T', and 'Y' have fewer instances, which could impact the statistical significance of the performance metrics for these classes.
  - Notably, class 'M' has only 3 instances, and classes 'E', 'O', and 'S' also have relatively low support, which might suggest either an imbalance in the dataset or a smaller representation of these classes in the test set.

7.2 Convolutional Neural Network (CNN)

7.2.1 System Infrastructure

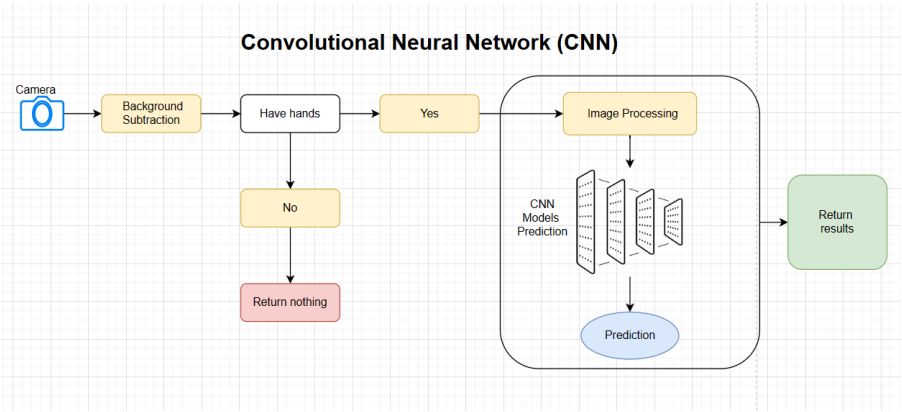

Figure 13: System Infrastructure of Type 2

7.2.2 Demo/interface

The following is a demo image that we have deployed, utilizing the rear camera. The system will make a prediction about the character based on the image of the hand placed within the predefined frame on the frame.

7.2.3 Evalution

Trainning

Classification Report
Based on the output:

1. **Overall Accuracy:** The model achieves an overall accuracy of approximately 99%, indicating its excellent performance on the test dataset.
2. **Classification Report:**

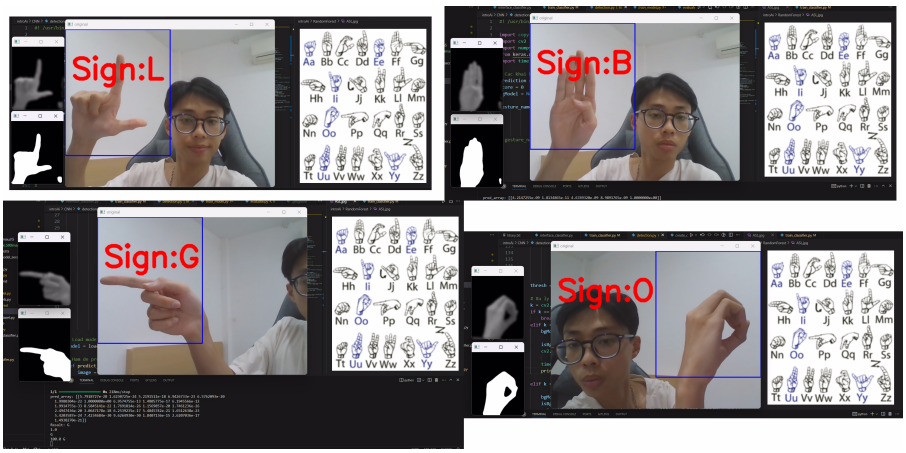

Figure 14: Interface of CNN Model

Epoch	Step	Time	Accuracy	Loss	Validation Acc	Validation Loss
1	325/325	59s	0.1300	2.9796	0.6900	0.9335
2	325/325	27s	0.6605	0.9740	0.8600	0.4472
3	325/325	28s	0.8337	0.4961	0.8965	0.3597
4	325/325	27s	0.8984	0.2902	0.9196	0.3632
5	325/325	28s	0.9246	0.2321	0.9227	0.3273
6	325/325	27s	0.9377	0.1884	0.9092	0.4110
7	325/325	28s	0.9472	0.1793	0.9146	0.3207
8	325/325	28s	0.9516	0.1439	0.9554	0.2216
...	...	...	...	...	...	...
27	325/325	27s	0.9845	0.0643	0.9665	0.2715

Table 3: Training Progress

- The precision, recall, and f1-score metrics are high for most classes, with values close to or equal to 1. This implies that the model effectively classifies these classes.
- However, some classes have lower accuracy such as class A, D, E, N, T, U, V, W, X, Y, Z, although they still perform well, further investigation may be needed to address potential issues.

3. **Macro and Weighted Averages:** Both metrics are high, with values close to 0.99. This suggests that the model performs well and is balanced across all classes.

In conclusion, this CNN model has been trained effectively and demonstrates high accuracy in classifying hand gestures on the test dataset. However, further evaluation on new datasets may be necessary to ensure the model's optimal performance.

7.3 Comparison

Let's compare the performance metrics between the CNN model and the RandomForestClassifier model mentioned earlier:

1. RandomForestClassifier:
   - Overall Accuracy: Approximately 99.41%
   - Classification Report:
     - Most classes have high precision, recall, and f1-score, with values close to 1.

Class	Precision	Recall	F1-Score	Support
A	0.96	0.98	0.97	100
B	1.00	0.99	0.99	100
C	0.99	0.99	0.99	100
D	0.98	0.97	0.97	100
E	0.95	1.00	0.98	100
F	0.98	1.00	0.99	100
G	0.98	1.00	0.99	100
H	0.98	1.00	0.99	100
I	1.00	0.95	0.97	100
J	1.00	1.00	1.00	100
K	1.00	1.00	1.00	100
L	1.00	0.99	0.99	100
M	0.98	0.98	0.98	100
N	1.00	0.96	0.98	100
O	1.00	0.99	0.99	100
P	1.00	0.98	0.99	100
Q	1.00	1.00	1.00	100
R	1.00	0.99	0.99	100
S	0.98	1.00	0.99	100
T	0.98	0.97	0.97	100
U	0.97	0.99	0.98	100
V	0.97	0.98	0.98	100
W	0.98	0.97	0.97	100
X	1.00	0.98	0.99	100
Y	0.97	0.98	0.98	100
Z	0.99	1.00	1.00	100
Accuracy			0.99	2600
Macro Avg	0.99	0.99	0.99	2600
Weighted Avg	0.99	0.99	0.99	2600

Table 4: Classification Report

  - Some classes have lower scores, indicating potential issues.

2. CNN Model:
   - Overall Accuracy: Approximately 99%
   - Classification Report:
     - Most classes have high precision, recall, and f1-score, with values close to or equal to 1.
     - Some classes have lower accuracy, suggesting a need for further investigation.

7.3.1 Comparison

- Both models demonstrate high overall accuracy, with the RandomForestClassifier slightly outperforming the CNN model by a small margin.
- In terms of precision, recall, and f1-score, both models perform well, with most classes achieving high scores.
- The RandomForestClassifier may have a slight advantage in handling certain classes, as indicated by its higher overall accuracy.

Overall, both models perform admirably in classifying hand gestures, with the RandomForestClassifier showing a slightly better overall accuracy. However, the choice between the two models may depend on factors such as computational efficiency, interpretability, and the specific requirements of the application.

D´i đây là ph`n Discussion đ̣c sa lại b`ng ti´ng Anh theo yêu c`u cua bạn:

8 DISCUSSION

8.1 Overview

The comparison involves two different approaches to Sign Language Recognition (SLR): a Natural Language-Assisted framework and a more traditional approach using Convolutional Neural Networks (CNN) and Random Forest Classifiers (RFC).

8.2 Methodologies

8.2.1 Natural Language-Assisted Sign Language Recognition (NLA-SLR)

- **Framework Components:**
  - *Language-Aware Label Smoothing:* This technique generates soft labels for training signs based on semantic similarities among glosses.
  - *Inter-Modality Mixup:* This blends vision and gloss features to maximize the separability of different signs.
  - *Video-Keypoint Network:* This novel backbone model processes both RGB videos and human body keypoints, leveraging different temporal receptive fields.
- **Performance:** Achieved state-of-the-art results on MSASL, WLASL, and NMFs-CSL benchmarks.

8.2.2 CNN and RFC for Hand Gesture Recognition

- **CNN Model:**
  - Achieved approximately 99% overall accuracy.
  - High precision, recall, and F1-score for most classes, though some classes showed lower accuracy.
- **Random Forest Classifier:**
  - Slightly higher overall accuracy at approximately 99.41%.
  - Similar precision, recall, and F1-score metrics as the CNN, indicating robust performance but also some issues with specific classes.

8.3 Comparison

8.3.1 Performance Metrics

- **Overall Accuracy:**

  - *NLA-SLR:* Demonstrates high performance across multiple datasets (e.g., 61.05% top-1 accuracy on WLASL2000).
  - *CNN and RFC:* Both models achieve near-perfect accuracy (99%), with RFC slightly outperforming CNN.
- **Class-Specific Performance:**
  - *NLA-SLR:* Effectively handles visually indistinguishable signs (VISigns) with either similar or distinct meanings, using language-aware techniques and mixup strategies to improve classification accuracy.
  - *CNN and RFC:* High precision, recall, and F1-scores for most classes, though certain classes show lower scores, indicating areas needing improvement.

8.3.2 Model Complexity and Computational Efficiency

- **NLA-SLR:** Uses advanced techniques like language-aware label smoothing and inter-modality mixup, which introduce additional computational steps but significantly enhance performance. The model also relies on pre-extracted keypoints, potentially increasing complexity but improving accuracy.
- **CNN and RFC:** Both models are relatively simpler in terms of architecture compared to NLA-SLR. RFC may offer better interpretability and computational efficiency, making it suitable for applications where resources are limited.

8.3.3 Applicability and Flexibility

- **NLA-SLR:** Designed specifically for SLR, making it highly specialized but potentially less flexible for other types of gesture recognition tasks. Its use of semantic information makes it robust for nuanced sign language interpretation.
- **CNN and RFC:** More generic models that can be adapted for various types of hand gesture recognition tasks, though they may require more data and fine-tuning to achieve optimal performance in specialized applications like SLR.

8.4 Table of comparison

Both approaches offer strong performance in hand gesture recognition tasks, with NLA-SLR providing state-of-the-art results in the specific domain of sign language recognition through advanced techniques. In contrast, CNN and RFC models, while slightly less accurate, offer robust performance with greater simplicity and potential adaptability to broader tasks. The choice between these models depends on the specific requirements of the application, such as the need for high specialization versus computational efficiency and flexibility.

9 Conclusion

Our innovative gestural communication tool represents a significant leap forward in breaking down communication barriers, particularly for individuals with hearing impairments. By seamlessly translating sign language into text or speech in real-time, we offer efficient and accurate communication solutions that empower users to express themselves effectively and be understood by others.

Criteria	NLA-SLR	CNN & RFC
Overall Accuracy	72.56% (top-1), 89.12% (top-5) on MSASL1000; 61.05% (top-1), 91.45% (top-5) on WLASL2000	Approximately 99%, with RFC slightly outperforming CNN (RFC: 99.41%, CNN: 99%)
Class-Specific Performance	Effectively handles VISigns with similar or distinct meanings, e.g., 90.49% (top-1) on MSASL100; 75.11% (top-1) on WLASL1000	High precision (0.99), recall (0.99), and F1-scores (0.99) for most classes; certain classes show lower scores
Precision	-	0.99 (Macro Avg)
Recall	-	0.99 (Macro Avg)
F1-Score	-	0.99 (Macro Avg)
Model Complexity	Uses advanced techniques like language-aware label smoothing and inter-modality mixup, introducing additional computational steps	Relatively simpler architecture; RFC offers better interpretability and computational efficiency
Computational Efficiency	Higher due to advanced techniques and reliance on pre-extracted keypoints	Lower, making them suitable for applications with limited resources
Applicability	Highly specialized for SLR, less flexible for other gesture recognition tasks	More generic, adaptable for various hand gesture recognition tasks
Flexibility	Robust for nuanced sign language interpretation	Requires more data and fine-tuning for specialized applications

Table 5: Comparison of NLA-SLR and CNN & RFC Models

With an anticipated accuracy rate ranging between 90% to 95%, our tool sets a new standard for precision in sign language translation. This high level of accuracy not only ensures the reliability of our system but also enhances the overall communication experience for users, instilling confidence and trust in the technology.

Central to our approach is the incorporation of cutting-edge technology, including advanced AI models such as the Random Forest Classifier and MediaPipe Hands. These state-of-the-art models enable us to detect hand gestures and interpret sign language with exceptional accuracy and efficiency, ensuring that our tool delivers optimal performance in real-world scenarios.

In addition to technical prowess, we place a strong emphasis on user experience, prioritizing intuitive design principles to enhance usability and accessibility for all users. Our tool is meticulously crafted to offer a seamless and intuitive interface that caters to the diverse needs of individuals with hearing impairments, fostering inclusivity and empowerment in communication.

As we continue to refine and develop our gestural communication tool, we remain steadfast in our commitment to creating a more inclusive world. We invite support and collaboration from stakeholders and partners to join us in realizing this vision, ensuring that individuals

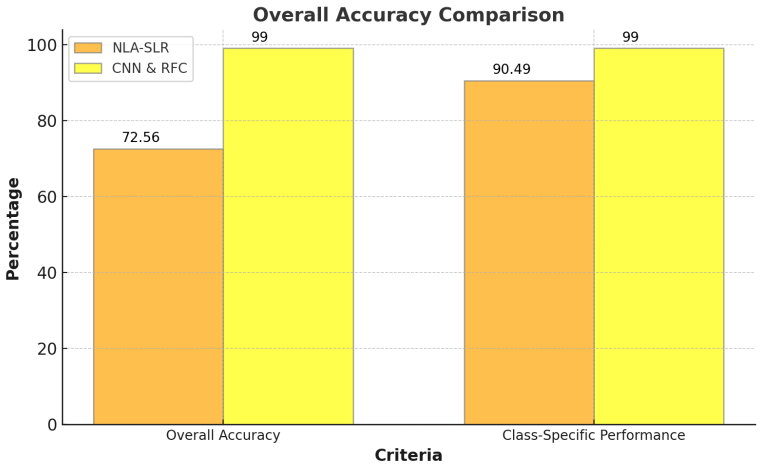

Figure 15: Overall Accuracy Comparision

with hearing impairments have equal access to effective communication tools and opportunities for meaningful engagement in society.

REFERENCES

[1] R. Zuo, F. Wei, B. Mak, Natural language-assisted sign language recognition, arXiv preprint arXiv:2303.12080 (2023).

[2] H. Alsolai, L. Alsolai, F. N. Al-Wesabi, M. Othman, M. Rizwanullah, A. A. Abdelmageed, Automated sign language detection and classification using reptile search algorithm with hybrid deep learning, Heliyon 10 (1) (2024) e23252. doi:https://doi.org/10.1016/j.heliyon.2023.e23252.
URL https://www.sciencedirect.com/science/article/pii/S2405844023104609

[3] GeeksforGeeks, Random forest algorithm in machine learning, https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/ (2024).

[4] Vitalflux, Different types of cnn architectures explained with examples, https://vitalflux.com/different-types-of-cnn-architectures-explained-examples/ (2024).

[5] Google, Mediapipe hands, https://github.com/google/mediapipe/blob/master/docs/solutions/hands.md (2023).

[6] GeeksforGeeks, Vgg-16 cnn model, https://www.geeksforgeeks.org/vgg-16-cnn-model/ (2024).

[7] Vgg-16 | cnn model, https://www.geeksforgeeks.org/vgg-16-cnn-model/?fbclid=IwAR3xiXTOvkLNCJPY81vAsIm3RPcwTQzkc1FqcEqNHnbXCpJhAnvwkjWeWa0, last Updated: 21 Mar, 2024.